AN EVALUATION OF PREPROCESSORS FOR NEURAL NETWORK SPEAKER VERIFICATION

Sheikh-Hussain Salleh

Thesis submitted for the degree of PhD
University of Edinburgh
1997

Abstract

This thesis is concerned with the application of artificial neural network models to speaker verification. In particular, the application of unsupervised and supervised neural networks with text dependent isolated digit systems is addressed. A hybrid neural network is proposed for speaker verification (SV). It consists of self organized neural networks and a network of multi-layer perceptrons (MLP). The basic idea in this system is the usage of vector quantization preprocessing as the feature extractor. The preprocessing stage measures local spectral similarities and uses this information to train the MLP. The benefits of this SV system are its speed and simplicity. The system was used in a series of experiments. The first experiment used a neural network model (NNM) with frame labelling performed by a client codebook known as NNM-C. Enhanced performance was obtained from this model when compared with the Hidden Markov Model (HMM) approach. The second set of experiments used the NNM with frame labelling from client and impostor codebooks known as NNM-CI. The results were poorer than when compared with the NNM-C and HMM. The amount of data, the choice of neural network parameters such as learning rate and momentum term and the use of hidden nodes are discussed.

In the third experiment a new approach used a speaker verification (HMM-MLP) method which combines a HMM based preprocessor with MLP. The output scores of the HMM were used as the inputs to the MLP. Compared with the NNM-C it is more successful since it has the capability of time alignment of the speech signals together with the efficient discrimination capability of the neural network. Moreover the method achieves better performance by the use of more than one feature set for each set of preprocessed parameters. It was found that further improvements in verification performance was achieved with the addition of the MLP and the HMM-MLP combination achieved 25% equal error rate reduction compared to the traditional HMM. Finally, the most important contribution of this research was the development and refinement of a NNM SV approach which incorporates a client barcode into the system design.

The conclusion from the experiments is that use of optimised preprocessors, especially HMM-MLP, yields improved results but this must be weighed against the increased complexity of system design.

Acknowledgements

I acknowledge sincerely with thanks the many contributions of all those who have helped in the preparation of this thesis.

Prof. Mervyn Jack and Dr. Fergus McInnes for their roles as my official supervisors and for continuous assistance and constructive criticism. Gratitude and utmost appreciation to them who read through the text of this thesis during its composition and suggested improvements.

Thanks are extended to my employer Universiti Teknologi Malaysia (UTM) and Public Service Department (earlier in my postgraduate career) for financially supporting this programme.

Thanks to Dr. Mark Forsyth for providing the starting point for my studies in the field of speaker verification; Dr. Ian Nairn and Mr. Bob Anstruther with regards to computer facilities. Of my many other colleagues - Dr. Mark Schmidt, Dr. Fabrizio Carraro and Mr. Keith Edwards who have helped to make my time spent in Edinburgh a worthwhile experience. To all others at Centre for Communication Interface Research and Centre for Speech Technology Research who had offered friendship, I extend my thanks.

My thanks and gratitude are extended to my postgraduate colleagues who have finished their studies or still studying at the centre. Maurilio, Pan and Nestor - the enjoyable moments that we shared shall never be forgotten.

All the individuals from the department of Kejuruteraan Elektrik, Unit Pembangunan Sumber Manusia, Bendahari (UTM) and Educational Services Consultants Ltd (Manchester) who kept all the affairs in order during my study.

Finally, the support of my family deserves my utmost gratitude. My mother (Maryam Mohd Noor) for her prayers and support throughout the project. To my wife and children (Hadrina, Hadri, Hasnur, Huzat) who have had to live with this thesis for years. I am deeply grateful for all your patience during this period in which I could not be as present in all of your lives as we would have wished.

Dedication

This thesis is dedicated to my wife Hamidah Ariffin.

Contents

List of figures

List of tables

Chapter 1

Introduction

Speech processing is a very important field for both practical and challenging reasons. Practically, speech processing will aid the disabled, improve productivity and change the way we run our lives. The goal of speech processing is to develop systems and techniques that enable computers to understand and communicate with humans. Thus, it poses a challenge for scientists and engineers alike. Speech processing has been successfully pioneered towards practical applications. The performance of these systems has improved greatly mainly due to advances in speech science and computer technology. The technology has improved the human-machine interface, both in terms of accuracy and user benefits. Specialised speech processing chips are also of major importance in the history of speech processing. These capabilities are now easily integrated into widely used applications.

Speech and language are a natural possession that humans have and creating a machine that can emulate this is a great challenge which is, as yet, unattained. The study of speech processing requires diverse scientific disciplines which include electrical engineering, computer science, linguistics, statistics, physics, mathematics, psychology, philosophy and biology. Among the more influential activities within these disciplines are work in pattern recognition, stochastic processes, digital signal processing, artificial intelligence, phonetics, physiology and acoustics. Research in this field has been going on since the 1950s. One of the most ambitious challenges is machine translation which requires a multidisciplinary approach as a vital component in the development of a solution. This does not mean that most of the complex problems of speech have been solved. Techniques developed in the early years are not directly extended to more sophisticated systems.

Voice data entry/retrieval for hands-busy or eyes-busy command and control application is particularly useful. Speech recognition plays a central role in recent medical applications in a hospital setting. The Arizona Heart Institute has focused on the development of a "paperless" intensive care unit where nurses can simultaneously monitor and report patient conditions using speech recognition. Speech recognition could also be used to aid the handicapped such as in the control of wheelchairs or in high technology application such as in cars or houses. Telephone based information retrieval in many commercial applications is important, especially in banking, movie schedules and phone billings. Speech recognition products can range from the sophisticated with large vocabulary and connected word recognition: to the low cost, with devices which implement simple small vocabulary isolated word recognition. In another application of speech processing, speech synthesis offers useful customer services such as voice messaging which can be accessed twenty four hours per day. Automatic speech processing techniques for identification of people by their voice characteristics have a number of interesting applications such as security, surveillance of communication channels and forensic applications.

Continuous speech recognition is more difficult than for isolated word as word boundaries are not easily detected in continuous speech. In addition there is greater variability in continuous speech due to stronger coarticulation. Another difficulty is the size of the vocabulary. The recognition accuracy of a system depends on the amount of training data available. Collection of sufficient training data can be practically difficult. In addition to vocabulary size issues there are other factors that can affect accuracy and robustness, speaker dependence and speaker independence. The speaker dependent recogniser requires the use of speech from the target speaker to train the model and will normally offer high recognition accuracy. Selecting a suitable model for a required task can play a significant part in determining the performance of the system. For example, for large vocabulary recognition, template matching techniques are untrainable and result in poor performance. For a classification task, training a single discriminative model can be more of an advantage then training several models representing each class. Another main source of difficulty is the inherent variability of speech. Words can vary widely in their acoustic characteristics from one occasion to another. In general none of the words have a perfect match. This form of variability affects word based recognition due to the non-linear extension and com-

pression of the timescale of a word from one utterance to another. Another contributing factor that can affect recognition performance is noise. This includes environmental noise, crosstalk, types of microphones and speaker induced noise such as lipsmacks, sneezing and clicks. It is not surprising to see a wide range of accuracy performance related to a given task. These factors are discussed in more detail in chapter 3.

In this thesis the research is concerned with the identification of people (mainly speaker verification) based on a connectionist approach. In particular the thesis focuses on the improvement of system performance obtained by certain developments of preprocessed speech signal inputs. Various performance measures are used to evaluate the automatic speaker verification (ASV) system for different aspects of performance. The results presented reflect an acceptable level of initial performance of the system with the assumption that speaker adaptation will be part of the implementation to handle long term trends in the voice. Chapter 2 provides an introduction to the field of automatic speaker recognition. The basic structure as well as classification of the speaker recognition systems are described. The types of sounds which contain the effective features for speaker recognition such as voiced sounds and the acoustics of nasality are described. A new direction of speech research using human hearing in the design of speech engineering is briefly described. Then, suitable choice of parameters used for the verification task is reviewed. This survey also includes speaker verification technology and its application in the telephone network environment. In recent findings with telephone based speaker recognition, performance degrades due to the mismatch of the training and test data. Researchers have addressed this issue and some results presented in the literature are described. The task descriptions for speaker verification such as fixed text, text independent and text prompted are described. Aspects of speech database requirements for developing this technology are also covered. Finally, an assessment of different algorithms used in speaker verification is presented.

In chapter 3, areas of particular interest for further research are identified. The main part of this chapter is devoted to consideration of the different preprocessor designs which are developed and implemented in this thesis. A brief description of a speaker verification system is provided. The challenges that the speaker verification (SV) poses to the neural network classifier are highlighted in this chapter. There are several factors which influence the performance of

different SV systems. In the remaining part of the chapters, such practical considerations are discussed with experimental evidence in support.

In the beginning of chapter 4 various approaches to hybrid systems applied in speech processing are described. The remainder of the chapter focuses on a hybrid approach based on stochastic and connectionist methods for speaker verification. The implementation of the stochastic model is based on (Forsyth *et al.*, 1993). The basic idea is to build a hybrid system which takes advantage of the two systems in order to enhance the performance of the verification system. The importance of the hybrid approach in reducing the complexity of the classifier system is shown with encouraging results.

Chapter 5 examines the viability of using a vector quantization preprocessor as a feature extractor. The usage of vector quantization preprocessor provides frame labelling from a client codebook and an impostor codebook. A series of experiments for each approach is conducted. The used of single codebook design provides a substantial reduction in inputs to the classifier for ease of implementation. Finally, the normalization results for different lengths of inputs for the classifier are reported showing that a relatively crude form of selecting the fixed inputs is also beneficial.

Chapter 6 describes further developments of the system which incorporate additional information for the classifier to perform the verification task. The new approach to the SV task involves the construction of a client barcode. The statistical technique used measures the quality of the match between two speech patterns. A number of preliminary experiments are carried out to assess the viability of the new approach. The results of the reliability of the client barcode with minimum overlapping samples are plotted to see if the similarity score gives an idea of the quality of the match. The results of these experiments suggest the extension of the system to include additional information of similarity to enhance performance. The application of the above approach to speaker verification is described and evaluated in the rest of this chapter. The distribution of errors over a range of speakers is examined and show considerable variation in performance amongst the digits and also large differences amongst the speakers in the distribution of errors.

Chapter 7 describes the different performance strategies in order to evaluate the performance of different SV systems. The SV approaches used in this thesis are compared with the traditional hidden Markov model (HMM) as well as the discriminating model known as discriminating observation probability (DOP) HMM (Forsyth, 1995). This chapter integrates the findings of chapters 4, 5 and 6. Comparison with different techniques of SV system include the use of speaker specific (SS) and speaker independent (SI) thresholds to calculate the equal error rate (EER).

Chapter 8, the final chapter, summarises the research findings. This chapter also presents some suggestions for future work which might be useful for further development and improvement to the techniques explored in this thesis. These include the possible use of genetic algorithms to select a subset of relevant input features with frame labelling from the client and the impostor codebook, the effects of handset variations and possible modifications to the development of the barcode.

Chapter 2

OVERVIEW OF SPEAKER RECOGNITION TECHNOLOGY

Past, Present and Future

2.1 Introduction

This chapter reviews the literature on speaker recognition using template matching, statistical models and connectionist models. In the application of speech processing using the above algorithms, there exists a large body of literature covering topics such as continuous word recognition, isolated word recognition, speaker dependence and independence system, automatic speaker and speech recognition. These papers will be discussed in detail later in this chapter.

This chapter also reviews briefly the basis of dynamic time warping (DTW), vector quantization (VQ), hidden Markov model (HMM) and neural network (NN) speaker recognition systems. The issues related to the construction of HMM recognition systems are discussed together with performance evaluations on large vocabulary databases. An alternative method that has been used to solve the speaker recognition problem is neural networks. Neural networks (NN) have been shown to perform the computation required by static pattern classifiers, vector quantization (VQ) and Viterbi decoding. Recently a new approach for a hybrid connectionist-HMM speaker

recognition system has been proposed. There are two approaches to the design of a speaker recognition system: the first approach uses multilayer perceptrons (MLP) as the probability estimator and the other approach uses MLP as labellers for HMM.

Experience with neural network design for specific applications such as speech recognition or speaker recognition has demonstrated the difficulty of selecting an appropriate functional structure for a network as well as its appropriate parameter values. There is currently no systematic methodology for determining the correct network architecture for a specific problem. The determination of an optimal network architecture is thus an open research problem.

We live in a computer age, in an information society. Computers have become the means of solving the many problems that we have encountered. Speech recognition and speaker recognition are two of them. There are two methods in solving the problem; knowledge based and pattern classifier. The latter is the more popular among researchers in this field. All systems use some kind of speaker model that characterizes the speech to be recognized. Speech is a natural form of communication for human beings. The use of normal speech is the simplest method to control instruments in a person's environment. With a given speech representation, pattern matching will be able to classify and detect the possible acoustic patterns, which can be words, syllables and phonemes from the speech signal. In the training phase, the features of the speech patterns for all the training words are stored as the reference templates.

In a speaker dependent system, templates are created from speech data from the specific speaker. The speaker has to utter a full list of the words in the vocabulary to train the system before it can be used for recognition. Although training can be an inconvenience to the user it is adaptable to the user and this improves the performance of the system. To recognize a new word, the new features are compared with all the reference sets. A matching score would be obtained to recognize the words. On the other hand a speaker independent system needs the utterance of the same words by several speakers to build the templates. This chapter reports

on research activities using template, statistical and neural network recognizers that perform the above operation.

Speaker recognition is a process of identifying the individual on the basis of the individual's speech characteristics. This technique will enable the system to verify the identity of the claimed person accessing the system based on the given speech information. The first part of the chapter will discuss the general topics and issues. Speaker recognition can be divided into speaker identification and speaker verification (SV). This thesis is devoted to discussion of the more specific topic of connectionist approach to speaker verification. The potential of connectionist models for SV is presented and the main algorithms are then reviewed. The respective performance and the potential of SV methods were compared to more conventional statistical approaches.

2.2 Principles of Speaker Recognition

2.2.1 Classification of Automatic Speaker Recognition Systems

The traditional method of using keys or personal identification numbers (PINs) for security purposes seems inadequate. This has brought about the use of biometric-based technologies. Biometric features are part of a person's physical makeup. Here analysis of the acoustic (speech) waveform is used to examine an explicit claim to a particular identity. Identifying an unknown speaker by any automatic means is termed as *Automatic Speaker Recognition* (ASR). It can be divided into two categories: *Automatic Speaker Identification (ASI) and Automatic Speaker Verification (ASV).* There has been a growing interest in ASV systems. A number of products have been developed such as hidden Markov model, neural networks, text prompted verification and combined use of speech recognition technology with speaker verification (Hunt, 1991)(George, 1995)(Perdue & Scherer, 1996). The progress of this technology will largely depend on the ability of speech technology developers to design reliable robust technology as well as the acceptance of this technology by the people.

Automatic speaker recognition methods can also be divided into text-independent and text-dependent methods. For text-dependent methods the same key words or sentences are used for training and testing. Text-independent methods do not rely on the specific text to be spoken. Text-dependent methods are usually template based and have a much simpler structure than text-independent recognition systems.

2.2.2 Basic Structure of Speaker Recognition Systems

Figure 2.1 shows the basic structure of ASI and ASV systems. The key elements of speaker recognition systems are feature extraction and classification. The feature extraction analysis computes a set of parameters from the speech signal. This initial stage is common to the learning as well as the test operating phases. Sometimes modification is done at this stage to extract the high level features aimed at representing a speaker with a limited number of features. These new features are compiled into a training and test data structure which is more economically represented. These data can then be used to build and test the speaker recognition model for the client speaker. The next stage will perform the classification which involves the comparison of the feature vector derived from the unknown speaker with the (claimed) reference vector. With a threshold set, the resulting distance above or below this threshold will determine the recognition decision.

Speaker identification is a process to determine the identity of the test speaker from the registered speakers using the specific information retained in the speech signals. This approach will identify the person accessing the system by means of comparison of the vector profile of the test speaker with each of the vector profiles of the speakers that make up the reference set. When the system makes a wrong decision, there can be two possibilities: a "closed-set" or an "open-set" identification. The first type of identification is when it assumed the unknown speaker is a member of the group of known speakers while the latter may not identify a reference model for the unknown speaker.

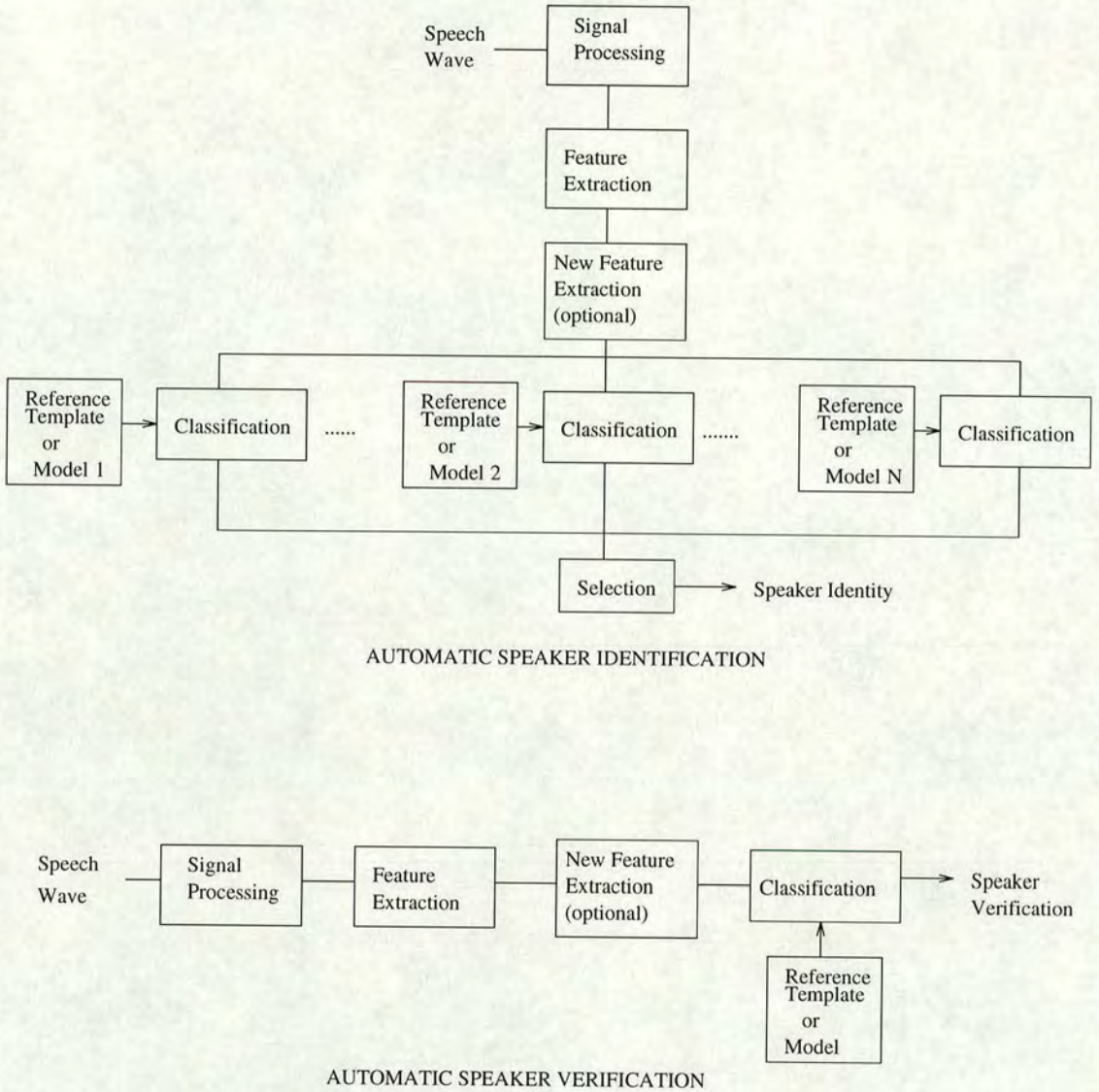


Figure 2.1: Basic Structure of Automatic Speaker Recognition Systems

On the other hand, establishing whether a speaker's claimed identity is correct by automatic means from the acoustics of his/her voice is termed as ASV. The operation of most speaker verification systems can be divided into two phases. In the training phase the reference sets for the speakers to be recognized are generated. During the verification phase, the claimed identity of an unknown speaker is verified by comparison with his test utterance. A distance score is obtained by comparing the test and the reference templates. If the distance (dissimilarity) is low enough, the speaker is accepted as being the genuine or true speaker. As an example of how to set the desired threshold value of user rejection and imposter acceptance is given in

(Rooney, 1990). The type of errors generated depends on the setting of the system's decision threshold. Errors can be classified in terms of falsely rejecting (FR) the genuine speaker and falsely accepting (FA) an impostor to gain access to the system. The effectiveness of a system is best described in terms of an equal error rate (EER). This performance measurment will be discussed in detail later in chapter 4 for the evaluation of ASV system performance.

The fundamental difference between ASI and ASV is the decision alternatives. In ASI the number of decision alternatives increases as the population size increases linearly while there are only two decisions to be made for the ASV task. The ASV system has only to reject or accept the unknown speaker regardless of the population size. Looking at this perspective this makes ASI a much more difficult task as performance decreases as population size increases. In either case when the unknown speaker does not match the model an additional threshold can be set to determine whether the decision is close enough to be accepted or if a retrial is needed (Furui, 1997). Several reviews of this field have already appeared such as (Atal, 1976)(Rosenberg, 1976)(Bennani & Gallinari, 1994)(Furui, 1994)(Furui, 1997).

2.2.3 Automatic Speaker Identification

Speaker identification and verification represent challenging tasks for researchers working in these areas. During identification, the test of an unknown speaker's data set is compared with the reference data sets using some form of distance measure. An important application of speaker identification is to identify people involved in criminal activity. The main players in this field are intelligence and law enforcement agencies. In addition there are systems that are being developed to extract specific information from highly structured conversations such as those between an aircraft pilot and an air traffic controller. There are a number of approaches in solving the problems of ASR. This section will describe approaches of ASI based on traditional as well as connectionist methods.

- Gish (Gish *et al.*, 1985)(Gish *et al.*, 1986) examined the effect of telephone channels in ASI. Multivariate Gaussian probability density functions were used to model the channel statistically. Cross channel problems can degrade the performance of ASI and they suggested new methods to improve performance.

- In the vector quantization approach each valid member has his own personalised codebook. To identify an unknown speaker the codebook with the minimum distance is found. Then the minimum distortion value is compared with the set threshold. Accepting or rejecting the unknown speaker will largely depend on the threshold value. Speaker identification systems have shown good performance results based on vector quantization (Soong *et al.*, 1985)(Buck *et al.*, 1985)(Xu *et al.*, 1989).

The remainder of this section will concentrate on aspects of the application of NN techniques to ASI. Neural networks can be used in conjunction with conventional ASR systems to extract features of speech. They can also be used to make decisions to verify whether two speakers X and Y are the same. Neural networks can be used for classification, or for clustering each talker's features.

- Oglesby and Mason (Oglesby & Mason, 1988) (Oglesby & Mason, 1990) have used the connectionist approach for automatic speaker identification. (Bennani *et al.*, 1990) used the LVQ algorithm to generate a codebook for each speaker. Their learning vector quantization (LVQ) based system was tested on 10 speakers. Two types of feature vectors were used, 12th order linear predictive coefficient (LPC) and 8th order Mel frequency cepstral coefficients (MFCC). They concluded that by mapping the speech problem to a pattern recognition problem a connectionist classifier can provide 97% accuracy. Both Oglesby and Bennani performed text-dependent experiments. The two approaches are simple and are applied small database problems.

- (Yin, 1990) reported text-dependent and text-independent experiments for ASI. The MLP used limited features for the test utterance and performance is slightly less than the conventional techniques such as the VQ codebook.

- Oglesby, Mason, Fredrickson and Tarassenko have reported results using radial basis functions (RBF). In both cases the RBF approach outperforms both a standard MLP and the VQ system (Oglesby & Mason, 1991) (Fredrickson & Tarassenko, 1994).

- (Bennani & Gallinari, 1991) reported a modular connectionist system based on time delay neural networks (TDNN). The system was tested on the TIMIT database using LPC parameters. The ASI system is capable of providing an average accuracy rate of 98%. This model has been tested on 20 speakers.

The published works on ASI have also included evaluation of performance for speaker verification. Having established the basic concepts of speaker recognition, further comments will be focused explicitly in the area of speaker verification which is the main topic of this thesis.

2.2.4 Automatic Speaker Verification

The first model for speaker recognition is speaker identification, as described in the last section. The second model is known as speaker verification.[1] Unlike speech recognition where the identity of the word spoken is the main interest, SV technology uses specific characteristics of the way in which an individual speaks to capture the voice profile of the individual. Factors such as use of good quality microphones, wide bandwidth and quiet environment have led to successful application of SV systems. The past decade has seen development of voice based services successfully implemented in the telephone switching network and there are now a varieties of services on offer to the public which include voice mail, bank transactions, data services and reservation systems. Access to some of these services such as computer networks, databases and proctected resources through telephone lines requires the use of personal authorization. So,

[1] This model is also referred to as speaker authentication or voice verification or voice recognition in the literature.

security is an important issue in these specific cases. In United States alone, companies lose US$1 million per day in data network break-in and US$3 billion per year for telephone fraud and there is no indication that these figures will reduce (Markowitz, 1997). Thus, the benefits of having personal identification or verification such as speaker verification as a reliable security option are self evident. This model, based on a person's voice is highly suited for telephone transactions, is easily integrated into the existing telephone network, has a simple user interface, is cost effective and can be easily integrated into existing software such as speech recognition and other security systems. This biometric technology is preferred by customers over other biometrics as well as non biometrics[2] (Markowitz, 1997). The customer's voice cannot be "lost or stolen" but its natural behaviour makes it more acceptable to the user especially when they are given the freedom of defining their own password.

It is important to have a standard for comparative evaluation of products and services of this technology. Standards are an indication of stability and promote confidence of this technology to the customers. In the present situation, SV in combination with word spotting technology seems to be well accepted by the public. The future for network based speaker verification is promising. However, there are central issues that need to be tackled first. For instance issues pertaining to signal variabilities, user training and adaptation, marketing channel and pricing and benefits to the user and the telephone industry.

The SV technology can reside at different points in the network with each having distinct advantages and disadvantages. This technology can reside at the subscriber premises or in the public switching telephone network (PSTN) or at service provider premises. The traditional method of receiving information through the PSTN requires the user to enter a digit string via a DTMF dial pad. The information obtained through this network can then be used to place an automated collect call or third party billing or banking transactions. This traditional approach is constantly abused by fraudulent callers. Speaker verification through the PSTN shown in Figure 2.2 as an excellent choice to combat this problem. The calls can be originated from

[2] This includes card, key system and PIN number.

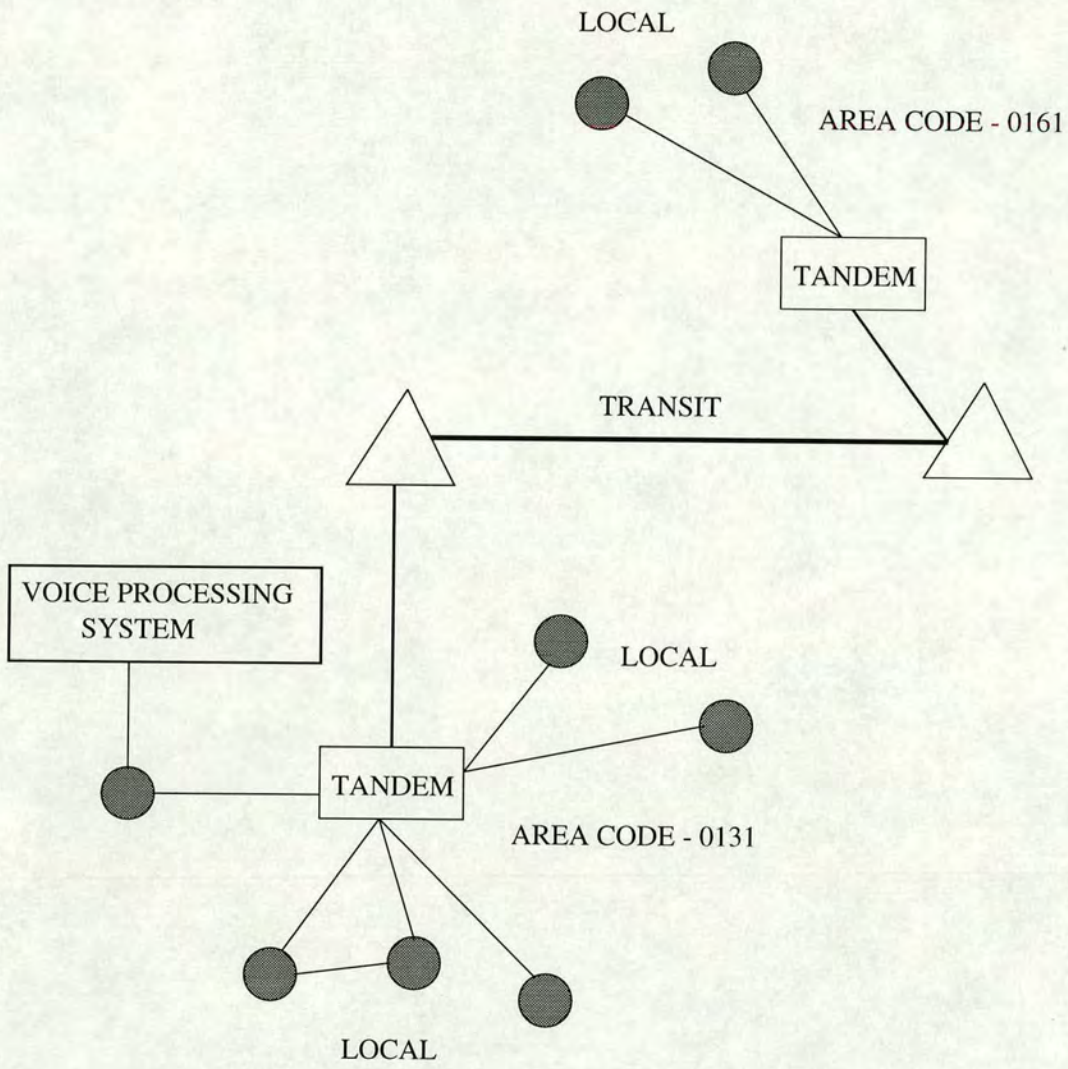


Figure 2.2: Speaker Verification Systems through PSTN

different area codes indicating different parts of the country. Calls can also be originated from the local calling area. These calls will be routed to the speaker verification system and an identity claim will be verified. In other words the switching exchange must be able to handle connections between its own terminations as well as the external junction. In long distance calls it is usual to transit calls between other exchanges and some of these exchanges exist solely for this purpose and have no termination on their own. As an example a call made from a subscriber can be originated from an area code 0161 and to be terminated to a local exchange belonging to an area code 0131 in which this subscriber can assess the voice technology. The voice processing system

will receive the signalling information, extract it and determine the type of service request by the subscriber. Once the call is established by the voice processing system the subscriber is normally prompted for a password. The voice password can be a several digit number selected by the caller. However, other types of text are also used for this application and this is discussed in more detail in Appendix B.

The voice processing system is normally linked by two buses. The speech bus will normally link the subscriber to the speech processing technology and vice versa where the system voice is sent back to the subscriber, while the process of monitoring and controlling is done through the computer bus link. The complexity of the system will depend on the architecture, software and hardware, the number of lines supported, disk storage availability and the speech/speaker recognition capabilities of the system. The system interfaces with the subscriber by replaying and recording compressed speech from the disk under the control of the voice service application processor. The subscriber will access the voice processing system via the telephone network interface as shown in Figure 2.3. To increase the efficiency of speech storage in the voice processing system, the caller prompt and inputs are normally compressed before storage and decompressed as the system interacts with the subscriber. As can be seen from the figure, various forms of speech/speaker processing are available in the system. The computer based text which is stored in the computer is converted to speech for transmission to the subscriber. One form of conversion is that in which the text is analysed and converted to phonetic units. The subscriber hears the voice after the phonetic equivalent is passed through a formant voice synthesiser. Currently available text to speech (TTS) converters are capable of handling 100,000 words correctly. However, although most TTS has an acceptable quality, there are still important challenges and research interest is growing in this field. Rapid development of new voices and the adaptation to specific tasks are open research fields. This involve different techniques blended into the TTS system. A general review of text to speech can be found in (Sagisaka, 1990).

DTMF tones are used to control the addressing and selection of menu options. DTMF tones can also be used to identify the subscriber. Speech recognition technology will be able to re-

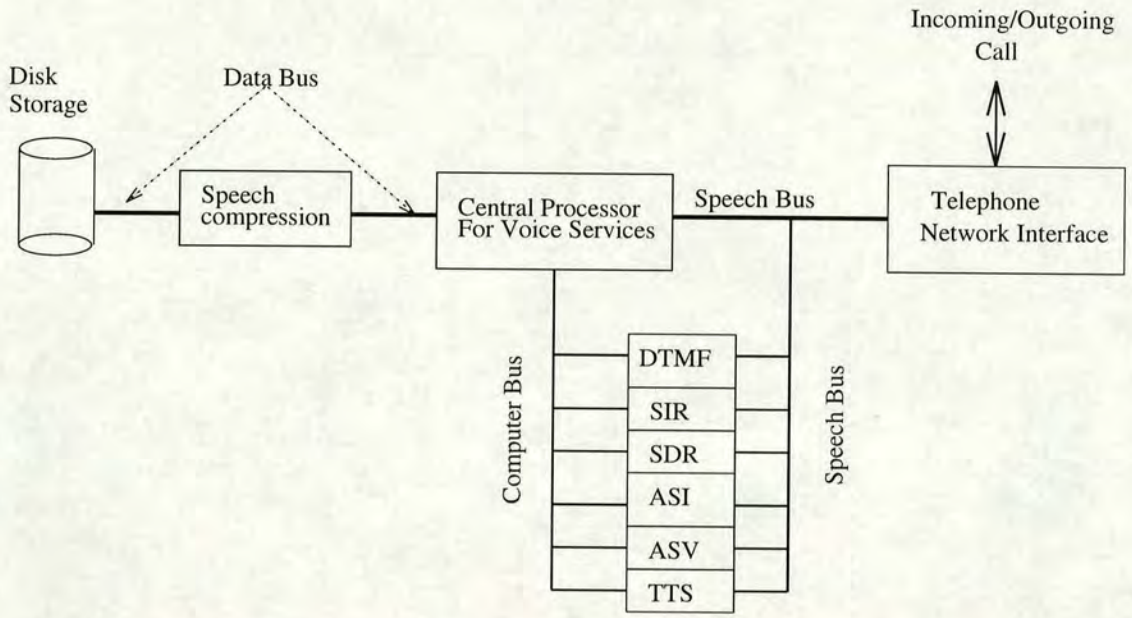


Figure 2.3: Speaker Verification as Part of the Voice Processing System Architecture

place the DTMF tones thus allowing a rapid growth in voice processing technology. Speech recognition can be divided into Speaker Dependent Recognition (SDR) and Speaker Independent Recognition (SIR). Isolated words and continuous words can be used for recognition. The SIR recognition is more complex than the SDR. The templates for the recognition process are generated in the off-line condition. Section 2.4 discusses in more detail speech recognition technology. For speaker recognition the voice print of the subscriber is taken during the training process and subsequently used to authenticate the subscriber's claim. The subscriber can be verified via the prompted words by the system. The speaker verification performance will have a high false rejection if the subscriber's voice is affected by cold or stress. The decisions deciding whether two speakers are the same have four possible outcomes. The outcomes are correct acceptance of a genuine speaker; or correct rejection of an impostor; or false rejection of the genuine speaker; or false acceptance of an imposter. Speaker verification technology as mentioned earlier is well accepted by the public, when perfected this feature can be used through the PSTN where secured financial transactions can take place. An overview of an operational voice processing system over the PSTN can be found in (Naik, 1994)(Ciria *et al.*, 1995).

The basic structure of a speaker recognition systems as well as a brief review of performances of ASI have been discussed in the earlier sections of this chapter. In the next section some of the fundamental aspects of the problem will be investigated. The performance of using different speech parameters for verification is discussed.

2.3 Techniques and Options in Automatic Speaker Recognition

2.3.1 The Nature of Speech

The human speech production system is a complex mechanism. Features such as height, weight, age and the structure of the vocal cords, nasal and oral cavities, teeth and lips play a major role in the speech production process. Speech is a series of sounds produced when air is forced out of the lungs through the vocal organs. These sounds are composed of a sequence of basic elements called phonemes which are governed by linguistic rules. The initial stage of generating speech begins in the lungs. Air is forced out of the lungs by muscle contraction into the bronchi and through the trachea. The air then is modulated by vibrations of the vocal cords which are located in the larynx. The larynx is involved in several functions such as respiration, swallowing and phonation. The larynx can be identified by the position of the thyroid cartilage. There are two cords of tissue stretched across the opening in the larynx. The frount ends of the cords are joined to the thyroid cartilage and the rear ends to the arytenoid cartilages. The arytenoids can be actively involved in the opening and closure of the airways and can be held so that the vocal cords are almost touching. If air is forced through the slit-like opening known as glottis, the cords will start to vibrate and so modulate the air flow. The vocal tract is an acoustical tube with non-uniform cross sectional area which begins at the vocal cords and ends at the lips. The vocal tract will change its cross sectional area at various different places as air flows through it. The frequency spectrum is shaped by the frequency selectivity of the tube. Detail derivation of the velocity transfer function for such a lossless tube can be found in (Parson, 1986). These resonance frequencies are called formants. The temporal variation of the cross sectional area of the sections of the vocal tract model alters the centre frequency, bandwith and the amplitude of

the separate resonances to produce different sounds. The process of the acoustic resonance is of prime importance in determining the properties of the speech sounds and further discussion of mechanisms and models of human speech production can be found in (Holmes, 1995).

In order to develop a successful SV system it is important to understand the nature of the speech signal and its relationship to a specific speaker. Speaker dependent information can be found in the vocal tract characteristics as well as in the segmental content of the speech. This speaker dependent information can be further classified into static and dynamic features. There are many different sources of speaker information in the speech signal that highlight the speaker's identity such as high level information and low level features. The high level information includes dialect, subject matter and style of speech which appears as pitch, speech rate and loudness. This type of information is not popular for automated techniques because of the practical difficulty of acquiring such information. It has been suggested that low level features which reflect the physical differences of a speaker's vocal tract and lungs are more reliable since physical characteristics cannot be altered at will and are relatively constant over time and vary greatly from speaker to speaker. This information appears as variations in formant frequencies, bandwidth, mean fundamental frequency and inclination of overall spectrum pattern. These two types of information are combined together in the speech signals making them difficult to extract. Normally, the individual features used for the verification task will contain both types of information. For more detail see (Doddington, 1985) regarding speech signals that convey information of the speaker's identity.

2.3.2 Feature Parameters for Speaker Verification

It is currently agreed by most researchers that feature extraction is one of the most crucial steps in the solution of practical SV problems. Before addressing the various speaker verification algorithms, it is appropriate to review the acoustical basis for SV. This section provides a brief description of the acoustic parameters which have been found to be helpful for speaker recognition. It also covers a description of different speech characteristics useful for distinguishing one

speaker from another.

2.3.2.1 Intensity

An important source of speaker information is the intensity of speech, also called gain. The variations in intensity of speech are caused by the variations of the subglottal pressure as well as the vocal tract shape which is a function of time. Speech intensity can be measured from the amplitude of the sound wave over a short time interval. In most of the cases for the SV task, speech intensity is always used with other parameters such as the fundamental frequency or pitch. For example, the mean and the standard deviation of the pitch and intensity along with the reflection coefficients were used in a SV task. The pitch was determined by the cepstral methods. In another study the spectral features were found to be the most effective, the pitch was the second most effective while the gain was the least effective (Markel *et al.*, 1977).

2.3.2.2 Pitch or Fundamental Frequency

Voiced sounds are produced when the vocal cords are tensed together and they vibrate as the air pressure builds up, forcing the glottis open, and then subsides as the air passes through. This movement modulates the air flow which is periodic or at least quasi periodic giving rise to a power spectrum with frequency components of several kilohertz. The fundamental frequency for an adult male ranges between 50 Hz to about 250 Hz with a higher frequency range for the adult female maybe up to 500 Hz. The speaker can control the pitch of the sound being produced. This he/she does by controlling the rate at which the vocal cords vibrate which then depends on the air pressure and tension in the vocal cords. The differences in fundamental frequency between different speakers and groups of speakers have been seen as having great potential for automatic speaker recognition.

In reviews by (Sutherland, 1989)(Rooney, 1990), studies have provided proof that pitch

contours could also be used for automatic speaker recognition. For pitch extraction, the zero crossing time domain technique was employed for the entire pitch contour of the sentence. It was found that the average value of the fundamental frequency provided better performance than the slope feature. A study by (Atal, 1972) involved the short time correlation analysis of the low pass filtered speech input to determine the basic fundamental frequency. (Luck, 1969) on the other hand had used pitch measures gained by cepstral methods. These studies have indicated that pitch on its own is a poor parameter to perform the speaker recognition task but if properly combined with other features will result in improvement of performance as can be seen in the discussion below.

Measurements such as pitch, intensity, formant frequencies and linear predictor coefficients are likely to be insensitive to the phase and/or spectral amplitude distortions found in the telephone exchange. Examples of such measurements are used in the Bell Labs system (Rosenberg, 1976) for automatic speaker verification. The measurements taken emphasise the variation of these features and are strongly correlated with prosodic behaviour of the speaker. For pitch and intensity analysis the input is low pass filtered at 900 Hz while the predictor coefficients or formant analysis is subjected to lowpass filtering at 3000 - 4000 Hz. Except for the predictor coefficients in which the measurement of the features is taken every 20 ms, all other measurements are analysed at 10ms interval. Dynamic programming is utilized for comparison of the sample and the reference contour. A distance is calculated between these two samples through a path of least accumulated distance. Using pitch and intensity an equal error rate of 3% was produced. An equal error rate of 1.5% was achieved when the predictor coefficients were added.

In a recent development, (Carey *et al.*, 1996) shows that simple parameters such as the mean and the variance of the pitch period in voiced sections of an utterance contain useful discriminative information. The initial estimate of the mean pitch was the average value of all the pitch values extracted over each 10ms frame. The estimate was refined by recalculating the mean pitch using only the samples of the pitch period found to be within plus or minus 35% of

the initial estimate of the mean pitch value. The process was repeated until successive iterations showed no change in the mean pitch value. The standard deviation of the pitch period was also estimated. In order to verify the unknown speaker, the pitch score is calculated based on the square of the distance between the mean for the unknown speaker's pitch and the average of the known speaker's speech weighted by the inverse of the target speaker's variance. Combining these features with the hidden Markov model the system is capable of achieving high performance results. The receiver operating characteristic was used to evaluate the system. The results indicate that these features are more robust than spectral features in handling the effect of channel distortions and noise. Identification studies which used the fundamental frequency means, variances and standard deviations include (Das & Mohn, 1971) and (Markel & Davis, 1979).

The complex nature of the pitch detection problem has brought about a variety of approaches for measuring that important parameter of the vocal tract excitation function. The most reliable (in terms of accuracy) methods are simultaneously the most sophisticated ones. The disadvantage of these approaches is that they require a great number of computations per sample of the speech signal and are also error prone. However, recently advances of speech coding have resulted in more reliable pitch detectors with less demanding computational effort (Patel, 1997). This could be well accepted for the developments in speaker recognition to further improve the performance of this task by the above method.

2.3.2.3 Predictor Coefficients

Among the most useful methods of speech analysis are those based upon the principle of linear prediction. This approach is important because of their accuracy and computational speed. Depending on the application in hand, one form of linear predictive coefficient (LPC) may be preferred over another. The are various forms of this parameter which have been used for this given task. These features are important in characterizing the spectral properties of speech in the

time domain (Makhoul, 1974)(Markel, 1974). The basic idea behind linear prediction is that the current samples are predicted using linear combinations of previous samples. By minimizing the square difference between the actual speech sample x_n and the linear predicted one, it is possible to determine the predictor coefficients. Typically 12 predictor coefficients are sufficient for speech band limited to 5 kHz. One can assume that over intervals of approximately 20ms the waveform is in fact stationary. The predictor coefficients vary as a function of time and are normally computed once every 20ms. According to (Atal, 1976) the linear predictor coefficients offer a convenient choice of parameters for speaker recognition for four basic reasons:

- they eliminate the necessity of selecting speech characteristics suitable for speaker recognition.[3]
- predictor coefficients represent the combined information of the formant frequencies, bandwidth and the glottal wave.
- being independent of pitch and intensity, addition of this feature will enhance the performance of the later model.
- can be conveniently implemented by digital hardware.

The short time spectrum has played a major role in speaker recognition. Specifically, linear prediction has been used extensively in speech recognition as well as in speech coding mainly because of its computational efficiency. Linear predictor parameters yield a set of predictor coefficients which are fewer in number than the spectral coefficients. The usefulness of linear predictor coefficients in automatic speaker recognition has been shown in earlier work including (Atal, 1974)(Rosenberg & Sambur, 1975). Other parameter sets related to linear predictor are the autocorrelation coefficients, cepstral coefficients, Partial Correlation coefficients (PARCOR coefficients) and the log area ratio coefficients. (Sambur, 1976) used these parameters as well as their mean value in a speaker recognition task. It was found that the log area ratio performed

[3] the speech characteristics might include the formant frequency or bandwidth or some property of the glottal wave

as well as the PARCOR coefficients and better than the predictor coefficients.

The principles of cepstral analysis involve the techniques of separating the excitation and vocal tract spectrum by the recursive manipulation of linear predictive coefficients or by inverse Fourier transform (IFT) of the log magnitude spectrum. The LPC analysis approach is based on short segments of speech with an all pole modelling constraint. This is a much faster approach and provides an accurate estimate of the speech parameters. On the other hand, the method involving the IFT based cepstrum (Owens, 1993) are derived based on the fact that the properties of the vocal tract tends to vary slowly with frequency while pitch tends to vary more rapidly and periodically with frequency. If $X(w)$ denote the spectrum of the voiced speech signal, $P(w)$ the spectrum of the pitch impulses and $V(w)$ the spectrum of the vocal tract. The relationship between the magnitude of these three spectra in logarithmic form can be expressed as follows:

$$\log|X(w)| = \log|P(w)| + \log|V(w)| \tag{2.1}$$

The contribution from $|V(w)|$ which is essentially determined by the properties of the vocal tract itself tends to vary slowly with frequency. The contribution from $|P(w)|$ tends to vary more rapidly and periodically with frequency. These two components can be separated by means of a linear filtering operation to produce what is known as the cepstrum signal. The slowly varying vocal function is represented by cepstral components near the origin while the contribution due to pitch occurs at multiples of the pitch periods. The horizontal axis of the cepstrum has the dimensions of time and is termed the 'quefrency' of the signal. Procedures for calculating these features can be found in (Rabiner & Juang, 1993). Thus the cepstral analysis can provide either spectral parameters or the estimate of the pitch (Owens, 1993).

A summary of results reported by (Atal, 1976) is given below. In this work there are 10 speakers and the sentence "May we all learn a yellow lion roar" with six reptitions were examined. In order to keep track of the signal variation, the speech signals were collected on two

different days 27 days apart. There were three separate recording sessions per day. The speech signal was band limited to 5 kHz. For each utterance it was divided into 40 segments of equal duration. A number of different parametric representations of speech were carried out in order to find the most suitable parameter for the speaker recognition task. The parameters studied are the predictor coefficients, impulse response of all pole filter, autocorrelation function, area function and cepstrum function of the impulse response. The cepstrum by definition is the inverse Fourier transform of the logarithm of the transfer function. For speaker identification score, the cepstrum function provided the best score while the area function provided the lowest score. For speaker verification, the speaker was verified if the distance between the test utterance and the reference was smaller than a preselected threshold. For the cepstrum function, speech durations of 1.0 sec provided better verification accuracy than for 0.2 sec duration.

(Luck, 1969) used the cepstral coefficients in a segment based study which also included the use of Fourier coefficients. The cepstral analysis used by (Furui, 1981a) in the SV experiment indicates that this feature is better than the log area ratio representation. The cepstral coefficients were derived from the LPC. The conversion from LPC to cepstral coefficients was carried out on a frame by frame basis. There were three polynomial coefficients representing the mean value, slope and the curvature of the contour. Furui also compared the cepstral coefficients derived from the IFT process. It was found that the IFT derived cepstrum gives similiar performance results but its implementation takes twice as long.

There have been other suggestions for improvements to SV systems especially in the use of multiple features. Soong and Rosenberg (Soong & Rosenberg, 1986) used both the LPC derived cepstrum and delta cepstrum for speaker verification experiments. Their findings also indicate that these features are relevant to characterize the spectral features of a speaker's voice. Discussions of these features and their performance results over dialled up telephone lines can be found in section 2.3.3. The delta coefficients from the cepstrum, (i.e. derivatives of the time functions of cepstral coefficients extracted at every frame period to represent the spectral

dynamics) have been accepted as an highly efficient parameter. Using two different codebooks, one for each parameter, the vector quantizer talker recognition was evaluated with 10 speakers. Two codebooks of size 64 were generated and used as a model for each speaker. An isolated digit database was used in this study with each talker registering 200 digits. The 200 digits were recorded within a two month period and divided into 5 different sessions. During classification of the text independent speaker identification, the unknown speaker's identity was chosen from the minimum accumulated distance among the N different speakers. The study also addressed several issues in connection with the short time spectral analysis in speaker recognition. The first issue considered was different representations of spectral information such as cepstra and delta cepstra. The next issue was the corresponding spectral distance and finally exploitation of the dynamic temporal structure of the speech sounds. Both of these latter features contain speaker specific information and the low correlation value indicates that the two features could be used to complement each other for speaker recognition. It was shown that the cepstra coefficients carry more speaker discriminative information than the delta parameters. However, the delta feature was shown to be more robust against transmission channel mismatch. In extension to the above work a database consisting of 20,000 isolated digit utterances from 100 talkers comprising 50 male and 50 female was used for full scale evaluation. The new speaker recognition system was similiar to the one reported above. The previous text independent system was evaluated with only a small amount of data. In SV mode an average equal error rate (EER) of 2.2% was obtained for the text independent mode and 0.3% EER for the text dependent mode on a 7 digit sequence test utterance. An important feature for this speaker recognition system as noted is that it can be easily extended from text independent to text dependent operation (Rosenberg & Soong, 1987).

The use of more than one feature set to improve the robustness of the SV has also been explored by (Forsyth, 1995). One disadvantage of this approach is that computational complexity increases significantly as more features are added into the system. The features used are mel-frequency cepstral coefficients (MFCC) and the difference coefficients of both the cepstra

and mel frequency cepstral coefficients. The MFCC feature set did not perform as well as the LPC cepstra.

Identification accuracy is a more sensitive indicator of the ability of a parameter for discriminating speakers. Consequently, an example of such a result in this section relates to a study in speaker identification. A by-product of the LPC analysis is the generation of a prediction residual e(n). The prediction residual signal e(n) is converted to an one sided autocorrelation sequence q(k). An algorithm was proposed (He *et al.*, 1995) to calculate a parameter from this prediction residual signal. This signal plays an important contribution since it carries information that has not been captured by the LPC coefficients. This new feature set calculated from the fast Fourier transform (FFT) based cepstrum q(k) and was defined as RCEP coefficients. Another feature set, LPC coefficients was calculated on short segments of a speech signal. A Hamming window of 32 msec was used in this analysis. The 16 LPC coefficients obtained from each frame of signals were used to derive 16 cepstral coefficients (LPCC). In additional to these features the pitch period was calculated for each frame of the signal as a feature. A speaker identification experiment was conducted to evaluate the effectiveness of these features. The selection of data was taken from the TIMIT database. There were 112 male speakers. There are 28,556 training vectors from 896 training sentences and 7,442 test vectors from 224 test sentences. It was found that prediction residual signals contain useful information for identifying speaker. Combining these features with a proper normalization procedure shows significant improvement in performance. LPCC features perform better than the RCEP but the overall best result is obtained with the combination of all the two features.

Several reviews based on the cepstral feature set as well as its transformation have appeared over the last few years. These include (Federico *et al.*, 1987)(Oglesby & Mason, 1988)(Furui, 1989)(Oglesby & Mason, 1991)(Gaganelis & Frangoulis, 1991)(Shrimpton & Watson, 1992)(Bennani, 1993)(Tsoi *et al.*, 1994)(Barger *et al.*, 1996)(Nakagawa & Markov, 1997)(Beaufays & Weintraub, 1997). It is clear from this that this feature is popular and useful for the SV

task. Through the above investigated work and comparison of several features it has been found that parameter sets related to linear predictor coefficients are highly efficient parameters containing speaker specific and linguistic information from short time spectral estimation of speech. Thus linear prediction provides an efficient way of representing the short time spectrum which is frequently used in speech analysis particularly in speaker verification. However, there are reports that performance of the speaker recognition system based on cepstral or linear predictor degrades when noise is present in the training or the test speech (Trent *et al.*, 1994)(Openshaw & Mason, 1994). Several studies to compensate this problem including cepstral mean subtraction, delta coefficients and RASTA filtering have been applied to speaker recognition and are discussed in the following section.

Different features have been investigated in SV as discussed above. The cepstral or the linear predictive cepstral feature is reported to provide good performance. Comparative tests with other features such as harmonic features, Line Spectrum Pair (LSP), Bispectrum features and Perceptually Based Linear Prediction (PLP) have shown to perform better or as well as the cepstral parameters. In a study by (Xu & Mason, 1989)(Xu *et al.*, 1989) instantaneous spectral information was represented by the PLP derived cepstra. Experiments on VQ speaker identification were carried out to evaluate the performance of the system between the PLP and LPC features with appropriate distance measure. The first 8 coefficients of the PLP cepstra carry information that is useful in the discriminating process. The PLP features gave better performance than the cepstra, delta cepstra or combination of these features. Liu and others studied different varieties of LSP for speaker recognition application (Liu *et al.*, 1990). The LSP is another alternative to linear prediction coefficients. The LSP features contain information regarding the variation of the glottis and vocal tract representing the speaker. This feature has also been found to be useful in the speaker identification task.

In the harmonic features representation the variations in the glottis and the vocal tract which determine the speaker's characteristics are transformed into the frequency domain. This

harmonic feature set is based on the harmonic decomposition of the Hildebrand - Prony line spectrum which has high resolution and accuracy. The use of harmonic features in speaker recognition consists of the fundamental frequency followed by amplitudes of several harmonic components. The fundamental frequency and the amplitudes of the first 19 harmonic components were used to form the harmonic feature vector. To evaluate the performance of this new feature the system was compared with 20th order LPC cepstrum features for phoneme based speaker identification/verification. Significant improvements in performance can be seen with the Harmonic features especially in the speaker identification case. For speaker recognition the harmonic features have performed better for isolated words and as well as the LPC cepstral features based on utterances of sentences (Imperl *et al.*, 1997)(Hayakawa *et al.*, 1997).

2.3.2.4 The Acoustics of Nasality

There have been many pieces of research conducted to determine the type of sounds which contain the most effective features for speaker recognition. Study has indicated that in general voiced sounds contain the best features. This research specifically on nasals is a part of a continuing effort to examine the benefits of this feature for speech and speaker recognition. In numerous studies long vowels and nasal features have been shown to contain discriminative properties of the speaker. The nasal tract is also a non uniform acoustic tube of fixed area and length. At the front end is the nostril and the rear is the velum which controls the acoustic coupling between the oral and nasal tracts. When non-nasalised sounds are generated the velum seals off the nasal tract while the velum is lowered and the nasal tract is acoustically coupled to the oral tract during the production of nasal sounds. During the production of the nasal sounds the oral tract is completely closed and the transmission path is only through the nostrils.

Experiments to determine the usefulness of nasal characteristics include studies by (Savic & Gupta, 1990) using the overall spectrum and by (Wolf, 1972) at extracting resonance feature. Savic and Gupta performed a text independent speaker verification system based on features of

vowels, fricatives, plosives and nasals. Each speaker was represented by a set of feature vectors derived from speech segments belonging to different classes of phonemes. Each speaker then trained a linear predictive hidden Markov model to obtain the different classes of phonemes. An improvement in performance was obtained by representing each phonetic category by a different model. The final verification score was a weighted combination of scores for individual categories. Since plosives do not contain much speaker information they were not included in the verification decision. Wolf showed that the use of nasal stops was no better than other segments such as vowels and voiceless fricatives.

Nasal cavities are generally recognised as being relatively fixed compared to vocal tract cavities which differ widely in size between speakers. Thus, there will be consistent acoustic differences between speakers based on the nasal feature. These factors necessitate the use of nasality in speaker recognition. However, the main disadvantage is difficulty in extracting the features. The complex spectral structure and the relatively low energy make it very difficult to obtain the formant features and to apply linear predictive analysis.

2.3.2.5 Human Auditory System

Humans are known to be capable of absorbing linguistic information during conversation. There is some evidence that several peripheral properties of hearing are at least partially responsible for the ways that speech has evolved and is used in human communication. Auditory model representations have also been applied to speaker recognition. Improvements in understanding of auditory modelling have led to extraction of auditory features (based on knowledge about human hearing) and training with a recognizer for the identification of speakers. Sound pressure waves that enter the human ear are converted into a sequence of electrical pulses which passes through the nervous system to the brain. There are three main regions in the ear which include the outer, middle and the inner ear. Vibration of the eardrum is caused by sound waves travelling through the auditory canal and impinging on the eardrum. In the middle ear there are three

small bones called ossicles which convey the vibrations of the eardrum to the oval window of the inner ear. The vibrations of the oval window cause pressure waves to propagate through the cochlear fluid. The pressure waves cause the basilar membrane to deflect. Attached to the basilar membrane is the organ of Corti in which hair cells can be found. These hair cells are in contact with the nerve endings (dendrites) of the neurons of the auditory nerve. These motions result in neural firings (electrical impulses) which are transmitted via the auditory nerve to the brain. Articles such as (Tobias, 1970)(Owens, 1993) offer further understanding of the human auditory system.

Over recent years speech researchers have used certain properties of human hearing in the design of speech engineering systems. An attempt to address the auditory model was carried out by Anderson and Patterson for speaker recognition using the self organizing feature maps (Anderson & Patterson, 1994). They used three different representations of the speech signals. The features are the mel-cepstral coefficients (MCC), the auditory image model (AIM) and Payton's auditory model (PAM). In AIM the spectral analysis is performed by a gammatone auditory filterbank. This filter converts the incoming wave into a surface that provides a representation of the basilar membrane as a function of time. A bank logarithmic compressors and a bank of adaptive threshold generators simulate the inner hair cells. These modules convert the basilar membrane motion that represents the pattern of activity at the output of the cochlea. On the other hand Payton's auditory model incorporates processing steps describing the conversion of the acoustic pressure wave signal at the eardrum to the time course activity in auditory neurons. This model converts the acoustic pressure wave into an action potential firing probability as a function of time through the stimulus. A two stage approach is applied to speaker recognition. The initial stage performs classification into broad class categories (39 phoneme categories) and the final stage uses one or more of those categories for speaker recognition. For speech recognition results, the auditory models perform better than the MCC. In the speaker recognition case AIM provided results comparable to MCC with PAM having the worst performance. A

summary of papers covering various topics [4] in relation to auditory model in speech recognition can be found in (Bourlard *et al.*, 1996).

At present there is little work being carried out base on the models of human hearing both in speech recognition and speaker recognition. The slow growth of the auditory model for the above tasks according to Bourlard, Hermansky and Morgan are due to testing of this task that do not highlight the weakness of the conventional feature extraction technique, ignoring the fact that not everything the hearing can do is necessarily used in human speech communication and finally failure to adapt the recognizer to the new feature properties. It was noted that the peripheral human auditory system can effectively integrate a rather large time span of the audio signal. Based on this fact they conclude that for auditory model the 10 ms of speech signals should not be use but using larger time span of the speech signal ranging between 30 and 200 ms for recognition. This is associated with the global properties of human hearing such as the short term memory of auditory periphery, firing rate adaptation constant or the forward temporal masking effects (Bourlard *et al.*, 1996).

2.3.2.6 Other Parametric Speech Representation

The above discussions illustrate the effectiveness of various speech characteristics used for automatic speaker recognition. The main focus has been given to those which are commonly cited in the literature review. This section discusses further valuable information about the performance of SV systems tested with other sets of parameters:

- Studies have shown that there is significant degree of correlation between short time spectra evaluated at different frequencies. In order for the system to be effective, stable evaluation of such correlations requires averaging over a long sequence of utterances with a minimum 30 s of speech (Atal, 1976). These features can be easily measured and are not

[4] Nonlinear frequency warping, root spectral compression, nonuniform spectral sensitivity of hearing, broad spectral integration and temporal properties of human hearing

suspectable to mimicry but they are easily affected by the varying frequency characteristics of the input channel.

- Formant frequencies can be defined as the resonance of the vocal tract and nasal cavities. The vocal tract can vary its shape which gives rise to different formant frequency values and different sounds which are speaker dependent. In continuous speech this formant frequency is undergoing constant change. The difficulty to this approach is to obtain reliable extraction and measurement of the formant frequency (Olive, 1971).

- It is a well established fact that people never speak words at exactly the same uniform rate. Sometimes words are spoken quickly and other times slowly. Thus the relative timing of different speech events in spoken utterances differs not only at specific occasions but also from one speaker to another. Doddington proposed a method of measuring such differences by determining the nonlinear distortion of the time axis of one utterance relative to that of another (Atal, 1976).

2.3.3 Telephone Quality Speech

Telephone quality speech poses much more of a difficult problem than clean speech for the SV task. This is because the telephone channel eliminates the higher frequencies in the speech signal which have been shown to have important discriminating information(Hayakawa & Itakura, 1994). Another important factor in telephone speech is that it is recorded over several sessions and on different handsets [5] which not only capture the relevant inter-speaker variation but also the unwanted intra-speaker channel variation. A telephone quality speech database can be from long distance or local calls; it can also be a payphone or in a quiet office; from various handset microphones and from cellular telephone calls. Researchers have also looked into ways to improve the robustness of speaker verification techniques against this distorted quality speech. As the handset and line vary from call to call, acoustic mismatch is liable to happen between data collected for training the model and testing. Mismatches and availability of the data severely

[5] The different types of telephone handsets available are carbon, electret and dynamic

affect the performance of the SV system. In addition, long term trends in voices can further accentuate the problem. Several papers have addressed these issues which have been tackled differently. Two types of normalization techniques have been applied: one approach tackles the acoustic mismatch based on speech features transformation, and the other is the channel handset adaptation models.

2.3.3.1 Transmission Conditions and Parameter Domain Normalization

Cepstral mean subtraction and delta coefficients have been applied to speaker recognition in order to reduce the effects of channel and handset mismatch as well as long term spectral variation. This approach is effective especially with text dependent speaker recognition with long utterances. In cepstral mean subtraction the cepstral coefficients are averaged over the duration of an entire utterance and the average values are subtracted from the cepstral coefficients of each frame. Such an example is discussed by Atal and can be found in the previous section 2.3.2. This approach is suitable for text dependent speaker recognition, however, it has to contain sufficiently long utterances or else some of the speaker specific features will be lost (Furui, 1994). The delta cepstra features tested by (Soong & Rosenberg, 1986) are found to be more robust to linear transmission mismatch compared to the cepstral features. Normalization techniques have shown the capability of reducing the long term spectral variation which is crucial for the performance of this system (Furui, 1994).

The effects of other parameters on verification performance have also been studied by Furui and Hunt. A review by (Rooney, 1990) found that fundamental frequency and gain parameters are less severely affected than spectral parameters. (Furui, 1981a) has attempted to compensate for the spectral changes by extracting some measure of the channel characteristics. (Hunt, 1983) found that formant frequencies had greater resistance to noise and non-linear distortions than spectral parameters.

Recent studies have shown that RASTA filtering applied to telephone quality speech can improve the performance of the SV system. Here, the assumption is made that the channel distortion is varying much more slowly than the speech signal. The short term cepstral average is calculated over an appropriate window for each cepstral vector and subtracted from that vector to provide the equalization. In one study (de Veth *et al.*, 1993) applied the RASTA filtering technique developed by (Hermansky *et al.*, 1991) to the cepstral coefficients. The results indicate a 25% EER reduction when this type of filter is applied to the speech database. In another study (Kao *et al.*, 1993) extensive experiments were performed on the KING database for the speaker recognition task. There were several features used which include interpolated SNR dependent cepstral normalization, bandpass filtering, RASTA-PLP, RASTA and bandpass filtering and RASTA. Using the modified RASTA called RASTA-PLP provides significant improvement. The best result is obtained from the combination of bandpass filtering and RASTA. The above techniques used are effective in eliminating the linear channel distortions but they are less effective in combating handset mismatch (Heck & Weintraub, 1997).

2.3.3.2 Handset Variations

Recently, however, it has been reported that significant classification performance degradation can be attributed to handset mismatch. Previous discussion applied different techniques to speaker recognition to compensate for linear channel distortions. There is research evidence attributed to mismatch of handset types which normally occurs because systems might be trained with a carbon button handset only but tested with electrets microphone handsets.

However, in a real application there is no control of the types of handset that the client speaker can use during enrolling. A more realistic approach is to train the system with different types of handsets. At Lincoln Laboratory, Massachusetts Institute of Technology, the handset TIMIT (HTIMIT) corpus and Lincoln Laboratory Handset Database (LLHDB) address the transducer effects on speech signals (Reynolds, 1997). These databases were designed to minimize all

confounding factors and produce speech that highlights the effects of the handset transducer. The HTIMIT database represents an artificial way of generating a large corpus of data with variations of handset types. It was constructed by playing a subset of the clean TIMIT corpus through nine telephone handsets and one Sennheiser high quality microphone. There were 384 TIMIT speakers (192 males and 192 females) with ten sentences. The handsets consisted of one cordless telephone handset, four carbon-buttons and four electret. This database is not ideal for real application assessment as the TIMIT corpus is played through a loudspeaker. Thus the HTIMIT corpus does not contain the effect of airflow (the plosive and fricative production has an effect on the handset output) from a person's mouth. To address this problem the LLHDB database is collected from 53 speakers recorded with the same type of handsets and microphone. The mel-scale cepstra and delta cepstra were used in the experiments. To minimize the linear filter effects the cepstral mean subtraction and RASTA were applied. For speaker identification, the average identification rate for same handset conditions was better than the cross handset conditions both for the male and female data. The overall performance was much lower using the HTIMIT corpus than the LLHDB corpus. The poor performance of the HTIMIT corpus is mainly due to the loose source transducer coupling and the interposed distortions from the loudspeaker. This represents an important study for better understanding of the distortions imposed by different transducers on the speech signal.

Such an approach was also used by (Naik *et al.*, 1989), where different handsets of five carbon, three electret and two dynamic were used to gather the speech data. The database consisted of utterances of four phrases from each of the ten handsets. Similiar to the above approach, a telephone interface which powered the telephone instrument eliminates any telephone channel distortions. There were 96,000 true speaker trials and 216,000 impostor trials. Improvement of results is seen with models trained with the handset database. The same effects could also be seen when handset adaptation model is included in the speaker recognition system (Heck & Weintraub, 1997). Instead of using random speakers to build the speaker recognition model, they focus on the handset mismatch problem by training separate handset dependent background

models. Their approach is different from the traditional model (Gish *et al.*, 1985) (state of the art Gaussian mixture model) which used random pooled background speaker normalization without paying any attention to handset types.

2.3.3.3 Telephone Applications in Noisy Environments

As mentioned before, channel effects caused by telephone lines as well as by handset variations seriously damaged the performance of the speaker recognition system. In some real applications, the speaker verification system is subjected to an adverse environmental condition. Speaker recognition with mobile telephones is gaining a widespread application. In this application an additional distortion source in the telephone network can come from the background noise. There are different kinds of noise sources that the telephone user can be exposed to which might include sources from a computer room, telephone booth, running car, exhibition hall or crowded environment. The distribution of these disturbances varies from the low level of a computer room or in a normal phone conversation to high volume noise in a car or crowded environment.

To adapt the speaker recognition model for noisy conditions, the HMM model as well as the neural network model has been successfully tried. (Reynolds & Rose, 1992) applied the Gaussian Mixture Model (GMM) to speaker identification under noisy conditions. Speaker model parameters are estimated in the presence of the background noise and a scoring procedure is implemented for computing the speaker likelihood in the noise corrupted environment. The performance of this system is evaluated with 16 speakers and this method is found to be highly effective in recognizing speech under a wide variety of interfering signals. In mobile voice communication, background acoustic noise imposes a great problem to the system. (Yang, 1993) applied a modified maximum likelihood estimate to suppress noise. In the approach given, both high and low SNR are considered in the system design. The speech detector proposed is robust and well suited to the rapidly changing background noise. However, the maximum likelihood

approach requires knowing the SNR which is difficult to measure exactly for non-stationary noise. (Matsui *et al.*, 1995) address this issue and show how to create a model for each speaker in order to compensate the noise based on HMM. The approach combines a speaker HMM and a noise source HMM into a noise added speaker HMM with a particular SNR. The likelihood normalization method based on a-posteriori probability was used in the speaker verification experiments. An a-posteriori threshold was set to equalize the probability of the false rejection and the false acceptance. The approach showed high recognition performance even when the SNR was unknown.

2.3.4 Speaker Verification Model and Threshold Adaptation

It is a well known fact that long term variability in the speaker voice will effect the speaker recognition performance. Normally the speaker model is updated to cope with the gradual changes in the speaker's voice. In many applications of speaker verification the amount of data available is normally constrained by what the client is willing to offer. The client may be reluctant to provide the large amount of data that is necessary to provide high performance. Thus, collecting many utterances at different sessions in a real life application can be difficult. An alternative approach is to build the speaker verification model based on limited training data collected at a few sessions and then updating the model using the collected utterances when the system is in use. Another important factor to maintain high recognition accuracy over a long period of time is the verification threshold. This verification threshold too needs to be updated constantly.

To investigate this issue, Furui performed verification experiments with speech data recorded over a long period of time. In the initial stage an optimum threshold was estimated based on the distribution of the overall distances between each speaker's reference template and a set of utterances of other speakers (inter-speaker distances). The inter-speaker distance distribution was approximated by a normal distribution. The threshold is calculated by its mean value and standard deviation (Furui, 1981a). This threshold is updated at the same time with the reference utterance. The verification experiments described above have also been carried out

by (Bernasconi, 1990). The experiment was carried out with speech data recorded over four months. In order to verify that the long term variation does affect the performance, the system was evaluated with a database recorded over three and a half years. There are only eight speakers used in this experiment. There were two thresholds in the system, one is the common threshold for all speakers and the other is a threshold specific for the individual speaker. The threshold adopted is similar to the one used by Furui. This approach does not take into account the intra speaker distribution since the amount of data available is small. It is much easier to estimate the inter speaker distance distributions by cross comparisons of the training utterances between different speakers. Even after three years the system is still capable of reliable performance. (Bonifas *et al.*, 1995) perform a text dependent speaker verification using dynamic time warping and vector quantization of line spectrum frequencies. The chosen method for threshold setting also follows that of Furui. The system was evaluated with 10 speakers over a period of 3 weeks. The advantages of this system, are that, as the thresholds and the reference for the DTW are updated automatically and regularly, the performance of the system will not decrease with time. The false acceptance rate is 0.085% without the key sentence and 2.2% with the key sentence.

2.3.5 Text Dependent Speaker Recognition Methods

The remaining part of this chapter is devoted to discussion of different kinds of speaker recognition methods which have led to interesting approaches and techniques. The fundamental idea of the text dependent approach is to find speech signals such as phonetic events that are peculiar to the speaker. These speech signals can be for example vowels. The recognition process is based on comparison of the speech signal patterns extracted from the phonetic events from an unknown speaker with phonetic patterns stored in the reference template. A typical approach to text dependent speaker verification is neural network based methods. For this approach, the neural network is not only trained by the client speaker's data but it also includes the impostors' data. This is closely related to the discriminative abilities of the neural network. Multi-layer perceptrons have shown good performance on several applications of speech and speaker recognition and below is a brief analytical description of this network.

2.3.5.1 Neural Network Based Methods

Pattern classifiers determine to which of the N classes an input pattern belongs. Pattern classification is often trained under supervision where the output is known. In the past few years neural networks have emerged as a powerful pattern recognition technique. Later in the section we shall see the many applications of neural networks in speech and speaker recognition systems.

Like any other pattern recognition techniques, neural networks act on data detecting how they resemble one another. Developing neural networks is not like writing software but rather they have to be trained. Most neural networks have parameters that influence their training process as they repeatedly see the training data. The choice of parameters then depends on experimentation and the results depend in part on the developer's experimental technique. Training neural networks is an emerging discipline that involves aspects of programming, statistics and signal processing (Hammerstorm, 1993). Nonetheless, neural networks can be difficult to train and in some cases will not lead to an applicable solution. Then, why do we use neural networks? Let us answer this question in a more general form in terms of its characteristics.

First, neural networks infer solutions from data without prior knowledge of the regularities in the data. Second, they can generalize, meaning they can recognize patterns that they have not seen before. Generalization is important because real world data are noisy, distorted and incomplete. Third, they are nonlinear and nonlinearity can be difficult to handle mathematically. Finally, they have parallel structures. Implementing them on parallel hardware will increase the computational rate.

There are differences between traditional classifiers, clustering algorithms and neural networks. When working with Dynamic Time Warping (DTW) using Von-Neuman computers the computations of the matching scores are done sequentially. Traditional algorithms have

serial inputs and outputs. If further improvement of the system is required then training may be complex and require large amount of storage and computation. However, different templates or HMM scores can be computed in parallel especially for isolated word recognition. Even though the input time series is processed sequentially for each model or template some speedup can be obtained by the use of parallel hardware. Neural networks as mentioned before have parallel inputs and outputs. Retraining the network for better performance requires no extra memory (Lippmann, 1988)(Huang *et al.*, 1988).

2.3.5.2 Back Propagation

There are different types of neural network, each having its own architecture, learning method and performance capabilities. A good example would be the back propagation (BP) algorithm. This particular algorithm is chosen because it's versatile and popular (Lippmann, 1987) (Maren *et al.*, 1990) and (Hush & Horne, 1993). Note that some other types of network may serve better than the given example, the key concepts apply to most neural networks.

A set of inputs labeled $x_1, x_2, x_3 \ldots\ldots x_n$, is applied to the artificial neuron as in Figure 2.4. This input can also represent the output of another neuron. Each input is multiplied by a corresponding weight, analogous to the synaptic strength of the biological neuron. All the inputs can be represented as vector X and correspond to the signals into the synapses. All of the weighted inputs are summed to determine the activation level of the node. The summation ($\sum$) portion corresponds roughly to the biological cell body with an output NET defined as

$$NET = XW \tag{2.2}$$

where W is the vector of weights.

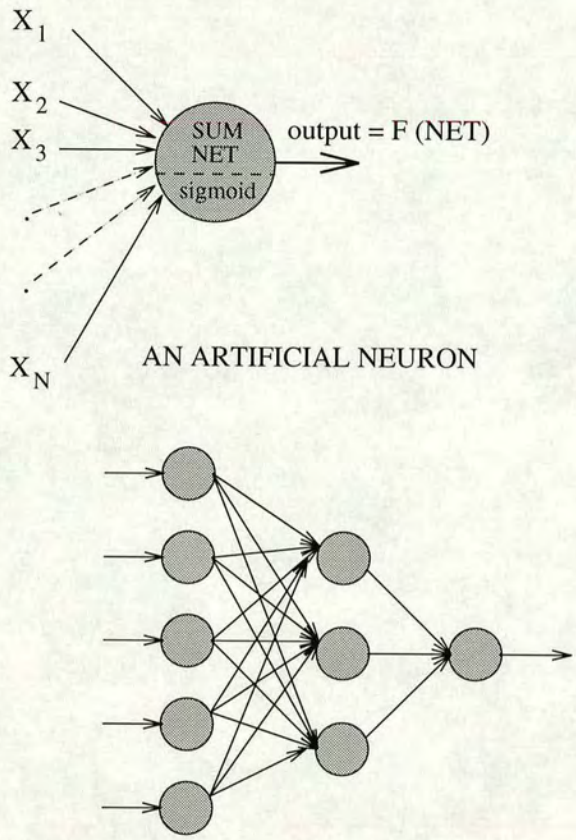


Figure 2.4: A Neural Network for Speaker Verification

The output of the NET signals is further processed by activation function F to produce the neuron output

$$\text{output} = \frac{1}{1 + e^{-\text{NET}}} \tag{2.3}$$

The type of function applied to this neuron is the sigmoid function. Many other functions (Hecht, 1991)(Eberhart & Dobbins, 1990) are applied to the neuron. Neural networks can be

taught to learn by supervised or unsupervised learning (self-organization). In BP, supervised learning is used where the output of the network is known. In unsupervised learning the network modifies itself in response to the inputs. The objective of training the BP network is to adjust the weights corresponding to the inputs to produce the desired output. For example, the input pattern of digit zero will be matched to the first output node with the target value set to one while the other remaining outputs are set to zero. The next input pattern (digit one) will be matched with the second output node and so on. Before starting the training process, all the weights are set to small random numbers. Training the back-propagation requires the following steps:

1. Select the next training pair from the training set and apply it to the network input.
2. Calculate the actual output of the network.
3. Calculate the error between the desired output and the actual output.
4. Propagate the error backwards to adjust the weights.
5. Repeat step 1 to 4 until the local minimum or the global minimum is reached.

The optimization technique that can be used to minimize the weight function F(W) is the steepest descent or gradient descent (Rumelhart & McClelland, 1986) method. After the training procedure a set of weights will be generated to be used later during the recognition phase. The patterns used for training are not used for testing the performance of the network. Peeling and Moore (Peeling *et al.*, 1986) (Peeling & Moore, 1988b) obtained good results for digits classification. Networks were trained using different number of layers and hidden nodes. Parameters such as the learning rate, momentum term and termination criteria play a significant role towards the development of the network architecture.

2.3.5.3 Application of Neural Networks to Speaker Recognition

Conventional speech recognizers using VQ techniques make use of a look-up table to speed-up the recognition process. This look-up table can also be performed by a single layer perceptron.

Neural networks can also perform vector quantization and a classic example is the Kohonen's feature map. Weights used in this map can be based on the Euclidean distance such as the k-means clustering. The auto-associative network is capable of reducing the dimensional inputs for speech processing. This network consists of the input layer, the hidden layer and the output layer. Usually, the number of hidden nodes is much less than the input nodes. Three layer feed-forward networks trained for autoassociative recall can be used for data compression. Neural networks have been applied to the problem of classifying static input patterns as well as dynamic classification of speech segments. The static networks discussed include the multilayer perceptron (MLP) and hierarchical neural nets that compute the kernel function. Dynamic networks on the other hand covers the time-delay multilayer perceptron as well as hierarchical networks and networks with recurrent connections.

2.3.5.4 Multi Layer Perceptron Based Methods

The connectionist approaches for Automatic Speaker Recognition are still relatively young (Bennani & Gallinari, 1994) in comparison to field of speech recognition. ASR is an important problem in the application of man machine interface. There are a number of approaches, in particular, the application of static and dynamic neural network. One of the important solutions to the ASR problem is the use of a multi-layer neural network. MLP applied to ASR have been reported to give good results but at the cost of excessive training times.

(Jou *et al.*, 1990) for example used a three layer perceptron network with 12 input units, 16 hidden nodes and 1 output node. A typical example of a neural network architecture for SV is shown in Figure 2.4. Each speaker inside the group has their own NN to verify the speaker. For the output node a 1 is the correct identity of the person and a 0 is the imposter. Input features to the NN are the LPC frames. 20 talkers were used to test the performance of the system. A closed test gives 98.5% while an open test gives 98.3% accuracy. The closed test indicates that the test tokens are used to train the NN and the open test indicates that the

test tokens are not used for training of the NN. This MLP is shown to perform non-linear discriminant analysis on speech frames. Reasonable performance results are achieved but at the expense of training time. In another study, a speaker verification experiment used 5 tokens from 70 speakers including the client speaker using MLP (Hangai *et al.*, 1992). For the features the locations and sizes of multipulses which were used for driving a LPC vocal tract filter in speech synthesis were clustered by a modified K-mean algorithm. The network size consisted of 90 inputs and a variation of hidden units from 1 to 5. The system design followed that of Oglesby and Mason where each speaker is represented by a MLP. The result is based on the receiver operating characteristics. In previous work carried out by this author, in order to get high speaker identification rates with these parameters, a large number of parameters is required and this leads to long identification times (Hangai *et al.*, 1990). For the verification study, the MLP with 5 hidden units shows the best performance result with 94% of correct acceptance. One of the most commonly referred papers and also possibly one of the earliest studies of the connectionist approach to speaker recognition is by Oglesby and Mason (Oglesby & Mason, 1988)(Oglesby & Mason, 1990)(Oglesby & Mason, 1991). The ASI problem was solved based on a feed forward neural network. This approach used data from the client and the impostors. By including these data, the MLP should be able to model the differences in people's speech. Oglesby and Mason also looked into the major issues of developing the MLP to suit this particular task. The two main issues that they have looked at are scalability as well as computation cost. Learning is done by minimizing the mean square error by the gradient descent algorithm. There are 10 speakers with 500 utterances were collected from them. 100 utterances were used for training while the remaining were used for testing. There are two types of architecture used. The first one is just a single layer in which the number of hidden units was varied between 16 and 128. The other network has two hidden layers, the first layer has 16 or 32 hidden units while the second layer varies from the number used in the first hidden layer down to 0.25 that value. The results indicate that a single layer MLP performs much better than the two layer network. The system performs as well as the VQ based approach with codebook size of 64.

2.3.5.5 Learning Vector Quantization Based Methods

In the above discussion, the algorithm relied on a supervised learning technique. Here, unsupervised learning called Kohonen self organising maps is discussed. The basic structure of the Kohonen map is shown in Figure 2.5. In other types of networks, all units adjust their weights in response to a training presentation but in an unsupervised learning, only one or at most a few units are allowed to adjust their weights in response to a input presentation. The output of a Kohonen unit is a weighted sum of its inputs. As seen with back propagation in supervised learning, the desired response of the network depends on the training input from the training class. For self organising maps two basic assumptions are made for this type of network. First, class membership is broadly defined as input patterns that share common features and secondly, the network will be able to identify the common features across the range of input patterns. The internal state of the network will be modified according to the unsupervised learning to model the features found in the training data. Kohonen, however, did use a supervised learning technique which he describes as Learning Vector Quantization (LVQ), trained with the target response. If the pattern matches that of the output unit then the weight connected to this unit is adjusted to move closer to the pattern, otherwise it is moved further away. The training procedure of this network can be found in (Morgan & Scofield, 1993). In a study carried out by (Bennani *et al.*, 1990), each speaker is represented by a codebook designed by the LVQ algorithm. The algorithm is based on nearest neighbor principles with fine adjustment through learning. The ASI system was tested on 10 speakers with ten sentences in French with each sentence duration between 1.5 to 3s. Two different features were used such as the LPC and the MFCC. It was found that the MFCC coefficients performed better than the LPC coefficients. The result based on the 10 speakers population with MFC coefficients provides an identification rate of 97%.

2.3.5.6 Recurrent Neural Network (RNN) Based Methods

In the previous discussion, most of the research focused on MLP and LVQ. These architectures are trained for direct classification and have been shown to be able to extract inter-speaker dis-

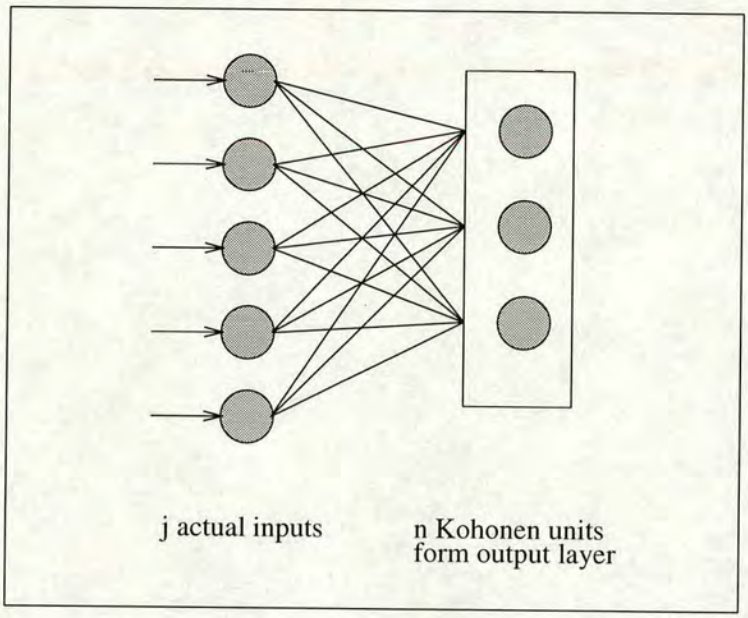


Figure 2.5: Kohonen Self Organising Model

criminant information. Another type of neural network that has shown promise for the particular task is the recurrent neural networks (RNN). Fundamentally, the RNN is the same as the MLP. However, the RNN architecture contains feedback as well as feed forward connections. The presence of this feedback enables the RNN to perform better than the MLP when the input speech is a dynamical time series. The RNN is trained through a modified standard back propagation algorithm. Investigation into RNN of this type was conducted by (Shrimpton & Watson, 1992). This type of network at any time instant depends on the states at all previous times. Time dependence is built into an RNN trained with recurrent back propagation by using the stored outputs from previous time steps as inputs at each time step. Different feature sets such as cepstra, delta cepstra and delta delta cepstra were used as the inputs to the RNN. This example illustrates a network with 30 input units, 20 hidden units and 2 output units. The transfer function used in all the units is the sigmoid function. The training data consisted of 36 templates from two groups while the test data consisted of 237 utterances of each digit from 34 speakers. The training scheme is the same of that Oglesby and Mason where each speaker has their own network. A comparison of the text dependent speaker verification system is made between two RNN architectures and two MLP. These architectures are RNN with memory, RNN with no memory, MLP with

memory and MLP with no memory. The overall performace indicates that MLP has the best performace but performs badly when memory is included. The EER rate for the RNN is about 7.5%.

Alternatively, there are other types of network that have been successfully applied to the speaker verification problem. In one approach, MLP networks working as auto-associators (Lastrucci *et al.*, 1994) are forced to produce the input to the output during the training phase. The rejection criterion is then based on the way the input is approximated by the output. The experiment was carried out with the DARPA-TIMIT database with 15 speakers. The LPC coefficients were used as the inputs to this neural network. Preliminary experiments were carried out to determine the optimal number of hidden units for the given speech. The best result was obtained with an architecture of 20-6-20. The network was trained to reproduce speech frames of the speaker under test presented at the input to the output layer. The effect of the auto-associator is that of reproducing on the outputs a "smoothed" copy of the inputs. During the verification stage, the test utterance from the unknown speaker is compared with the correspondent output vector. Adequate threshold criteria are proposed for performing rejection. This auto associator method is seen as an adequate approach for speaker verification. There is also development of SV motivated by the function of the brain which is able to receive new pieces of information as they arrive without changing the stability needed to ensure that the existing information is not erased or corrupted in the process. Adaptive resonance theory (ART) architectures are neural networks that self organize stable recognition codes in real time response to arbitrary sequence of input patterns. Such networks were constructed by Yegnanarayana and others for the SV task (Yegnanarayana *et al.*, 1994).

The above NN suffer from poor performance when applied to large problems. This is specifically true with the MLP. Back propagation learning is not readily applicable to problems which require large number of units and more layers. Another drawback of the above approach is that the complexity of the system increases as the number of talkers increases. Modular connectionist systems have been applied to solve this problem. Other popular networks that have been applied

to speaker recognition include time delay neural network and predictive neural networks. There will be more discussion of such networks in section 2.3.6 of this thesis.

2.3.5.7 Hidden Markov Model Based Methods

Hidden Markov Model (HMM) is a probabilistic technique in a time series. The technique uses a stochastic method. A hidden Markov model is a collection of states connected by transitions. Each transition can be divided into two components: a transition probability to mark the Markov chain and second the output probability for the output symbol. Figure 2.6 shows a simple left to right hidden Markov model.

The simplest of the HMMs is the discrete type. Since it is a left to right model no 'backwards' jumps are allowed. The three circles represent the states of the model and at a discrete time t, the model makes a transition between states and emits an observation. The probabilities of moving from state to state are tabulated in an N x N state transition matrix, $A = [a_{ij}]$. The probabilities of emitting symbols are tabulated in an N x N matrix $B = [b_{ij}]$.

$\{s\}$- A set of states including an initial state S_I and a final state S_F.

$\{a_{ij}\}$- A set of transition where a_{ij} is the probability of taking a transition from state i to j.

$\{b_{ij}(k)\}$- The output probability matrix: the probability of emitting symbol k when taking a transition from state i to state j.

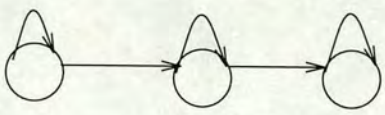


Figure 2.6: A Simple Left to Right Hidden Markov Model

2.3.5.8 The Three Problems of HMMs

Given the form of Hidden Markov model discussed above there are three main problems that need to be solved in order for the model to be useful in real application. The problems are:

1. The Evaluation Problem- Given a model and a sequence of observations, how do we compute the probability of the observation sequence.
2. The Decoding Problem - Given a model and a set of observations, how we choose a state sequence that produced the observations.
3. The Learning Problem - Given a model and a set of observations, how we adjust the model parameters to maximise the probability of the observation sequence.

Solutions to these three problems can be found in (Rabiner, 1986) (Cox, 1988)(Lee, 1988). Consider the design an N-state HMM for each word for a V word vocabulary. Using the vector quantization technique the symbols are derived from an L size codebook . During training the solution to problem (3) is used to optimally compute model parameters for each word model. The solution to problem (2) will give the best matching state sequence given an observation sequence. Lastly, to do recognition of an unknown word solution (1) is used. The word whose model gives the highest score is the recognized word. HMM have several interesting properties (Lee, 1988):

1. They requires minimal supervision.
2. They have solid theoretical basis.
3. They scale gracefully to increased training, requiring only linearly more computation.

The three main forms of HMM are discrete (DHMM), semi-continuous (SCHMM) and continuous (CHMM). For the DHMM, vector quantization is performed on the speech data to obtain the desired codebook (Gray, 1984) (Pan *et al.*, 1985). Under the DHMM framework, VQ is first used to obtain the output symbols of the input data. DHMM then models the observed symbols. In contrast, the CHMM uses continuous mixture probability density functions to model

speech parameters directly without using VQ, and usually needs extensive training data and computation times. In SCHMM, also called the tied mixture model, the VQ, the discrete HMM and the continuous mixture HMM are unified. SCHMM has the ability to model large mixture probability density functions. The number of free parameters and the computational complexity can be reduced because all of the probability density functions are tied together in the codebook. Detailed description of the three main forms of HMM can be found in (Huang *et al.*, 1990).

HMM have been applied to speech problems since the 1960s after the discovery of a method for optimizing the parameters of the Markov model to match observed signal patterns. However it is in the 1980s that we see a lot of development of speech recognition system based on HMMs. An example by Rabiner (Rabiner *et al.*, 1983) shows an approach to speaker-independent, isolated word recognition in which the vector quantization and hidden Markov models are combined to form a speech recognition system using a vocabulary of 10 digits. The front end uses linear predictive coding for feature extraction of speech. Several factors that affect the performance of the system are also discussed in this paper. The first factor is that constrained serial models perform better than unconstrained models. A second factor is that more training data yields better estimates of the HMMs parameters. A third factor that can be seen is that HMMs with 5 states should be used for each word as more states do not lead to further improvement in performance. A fourth factor is that a single model for each word is adequate (effects of different random start on the overall error performance are small) and finally there is no advantage using a parallel HMM structure.

2.3.5.9 Application of Hidden Markov Models to Speaker Recognition

Since there are a wide variety of services which require the use of telephone, accurate verification capability over the telephone could lead to more commercial application of this technology. Speaker recognition based on hidden Markov models (HMM) has been the subject of active research for many years. The success of this approach has brought about many potential appli-

cations. For speaker recognition, a speaker dependent hidden Markov model for a client speaker is normally trained with the enrolment data in one of the training sessions. Under such cases, the model matches the probability density function of the training data perfectly. During the verification process, the test data are normally subjected to various different conditions which seriously affect the performance of the speaker recognition system. The test data are often collected in different sessions, with different telephone channels and handsets. Since the acoustic test data is different from the enrolment session it usually causes mismatch between the test data and the trained HMM. Several analytical approaches have been applied to this model in order to tackle the mentioned problem to the task of speaker recognition.

HMM based methods have the capability of modeling statistically variations in spectral features. Therefore, ASV based on HMM have achieved significant recognition accuracies (Naik *et al.*, 1989). Two different types of database were used in the experiments. These databases were intended to measure the variations caused by the handsets as well as the telephone channels. The speech signals are transformed to LPC features. Using the same database the HMM system was compared to speaker verification based on DTW. Two types of single word HMM models were implemented, in the first model all states have the same covariance matrix and in the second model each state has its own covariance matrix. The second model is a discriminative model. This speaker discrimination model uses a linear transformation matrix that is designed to discriminate between the true client speaker and the impostors. This model is seen as a better choice rather than creating a set of statistically uncorrelated features, since these uncorrelated features may not still be a good discriminant. The results show that including the variations of handset types and channel variations provides better performance. Speaker verification discriminating model provided a 90% improvement over the SV DTW and 50% over the non-discriminant HMM model.

A more efficient way to represent speakers is the tied mixture approach (de Veth & Bourlard, 1994) rather than the discrete HMM. Here, the single Gaussian HMM and tied multi Gaussian

HMM were used in a SV task. An important characteristic of the tied mixture model is that it is capable of attaining acceptable verification performance when applied to the text independent mode. The LPC coefficients were computed from the speech signals and transformed into 12 cepstral coefficients. The HMM system was based on digit strings, and a phonemic approach was introduced into the system. To reduce the number of parameters, context dependent phoneme models were represented by single state HMMs described by single Gaussian probability density functions. The means and the variances describing the Gaussian probability density functions were estimated using the Viterbi algorithm. With enough training data the tied mixture models perform better than the single Gaussian approach.

In another approach, subword units were represented by HMM. The two types of subword units are phone-like units (PLU's) and acoustic segment units (ASU's). The extraction of the ASU's parameters was taken directly from the acoustic signal without use of any linguistic knowledge while PLU's are based on the phonetic transcription of spoken utterances. The two different features used are cepstral and delta cepstrum features. The verification performance has been evaluated on a 100 talker database of 20,000 isolated digit utterances. Good performance can be obtained using these subwords for isolated digit recognition. The results of the experiments also show that there are only small differences in performance between PLU and ASU based representations (Rosenberg *et al.*, 1990). In summary, an EER of 7% to 8% is achieved for the 1 digit test utterance and 1% or less for the 7 digits test utterances. Further improvement of the system was achieved when an adaptation technique was introduced into the system.

At the University of Edinburgh there is on going research to improve the performance of automatic speaker verification systems. Performance of speaker verification system varies greatly with the amount of training data. One of the objectives of speaker verification is to obtain high performance with limited amounts of training data. In the experiments carried out average equal error rates of 14% (DHMM) and 12% (SCHMM) were achieved for single isolated digits and 4% (DHMM) and 2% (SCHMM) for a sequence of 12 isolated digits for the 10 training

tokens models (Forsyth *et al.*, 1993). In another study for speaker verification (Rooney, 1990), examines the suitability of nasal resonance patterns. Equal error rates of 12.8% (for 15 males) and 13.8% (for 14 females) are achieved with this specific parameter. The above approaches, especially the hidden Markov model, will be discussed in more detail in the succeeding chapters. Also at the University of Edinburgh a real time speaker verification system using hidden Markov models was built using a signal processing card as the front end, a transputer for parallel HMM probability estimation and the IBM PC/AT to control the overall process. Further descriptions of this system can be found in (Logan *et al.*, 1990).

2.3.5.10 Dynamic Time Warping and Vector Quantization Based Methods

The HMM technique can be compared with another well established technique for speech recognition using template matching. The basic idea behind template matching is that each word in the vocabulary is represented by a template. The template is the reference pattern created from speech data. During recognition, the unknown input words are compared with the reference templates. The input word which best matches the template is the correct word. Since people rarely speak words at exactly the same uniform rate, some method of time alignment is required in order to compare the test pattern with reference word patterns. A time alignment matching algorithm (Myers *et al.*, 1980) (Ishak *et al.*, 1992) that can handle this problem is dynamic time warping (DTW). During recognition, the DTW will nonlinearly expand or contract the time axis to match the phonemes in the training and test sets. This is necessary in order to eliminate inconsistencies in the speech signal. The dynamic programming technique will map the time axis of the test patterns onto the time axis of the training patterns in such a way that the resulting dissimilarity is minimized. By warping the time axis of one speech pattern, the timing differences between the two speech patterns can be eliminated. Part of the system design will include the constraint problem of DTW (Myers *et al.*, 1980). Often these constraints will affect the system performance. In HMMs, the DP matching procedure is replaced by computation of the probability of the speech data given the model. The advantage of this is that the local

constraint function can be re-estimated by an iterative process and this allows optimization on the training data base.

It is known that short term training feature vectors contain important speaker characteristics. However, if the number of training parameters is too large then it is impractical since the amount of memory as well as the computational requirement of the system will also grow. Another important method that has contributed successfully in this field is Vector Quantization (VQ) which is an efficient way of compressing the training data. Speaker specific features are stored as elements of a codebook. In the actual process of VQ, the input vector for the unknown speaker is compared to all of the vector centroids and the one with the smallest distance is chosen. The cumulative distance is compared to a specific threshold to make an acceptance or rejection decision.

2.3.5.11 Application of Vector Quantization and Dynamic Time Warping to Speaker Recognition

(Furui, 1981a) implemented speaker recognition experiments using the typical structure of a DTW based system. LPC coefficients were used in the experiments. Spectral equalization was applied to each spectral coefficient in order to compensate for transmission distortion and intra speaker variability. The minimum distance was calculated based on the time function of the reference and the unknown template. The overall distance score was compared to the threshold for the verification decision. In another approach by Furui, statistical and dynamic features were used and the system was evaluated under a long term strategy. Statistical features were extracted from long term averaged spectra of a sentence utterance and time averaged characteristics of log area ratios and fundamental frequency were derived from the voiced portion of the spoken words. On the other hand, dynamic characteristics have been analysed by the use of time functions of the log ratios and fundamental frequency. The warping function is calculated in the course of registration. In the statistical case, the mean value and the standard deviation for each time function and a correlation matrix between these functions are calculated in the voiced portion

of each word and after a feature selection procedure. In the case of dynamic features, the time functions are brought into time registration with reference functions. Only a slight difference in performance can be seen from the results. The calculation involved for recognition using the statistical approach is only one tenth of that using dynamic features. It is thus more economical to use the statistical features rather than the dynamic features. Applying spectral equalization to the statistical features, the performance of the system was evaluated five years later from the training time and provided 96.3% verification accuracy (Furui, 1981b).

(Rosenberg & Soong, 1987) developed a system based on constructing highly efficient short term spectral representations of each client using VQ codebook construction techniques. The approach used was text independent but the system can be extended to text dependent for better performance. Larger amounts of training data are required in the text independent mode to take into account the higher acoustic phonetic variability. The recognition procedure for the text dependent case is different from the text independent. The text dependent mode normally compares the speech events of the reference and the test utterance by non linear time alignment while the text independent case does not use time alignment. The text independent case normally involves statistical averaging. Both text dependent and text independent experiments were carried out by Rosenberg and Soong. The system was evaluated with a 100 speaker database. The average EER for the 7 digit sequence for the text dependent speaker verification system is 0.3%. There was 86% improvement using the text dependent mode over the text independent.

2.3.6 Text Independent Speaker Recognition Methods

The above discussions have addressed text dependent speaker recognition methods. However, if the system has a fixed text it is likely to be defeated by recording of the person's text. So, some systems will change the required text or sentences every time the recognizer is in used. Text independent speaker recognition is much harder to perform because of the freedom of the text available in the system thus causing high variability of speech related to this text. Here, some work that addresses the speaker recognition problem is presented within the framework of

text independent recognition. The following discussion includes different kinds of methods that have been investigated for this specific task.

2.3.6.1 Neural Network Based Methods

In the previous sections, NN techniques can be seen to closely approach the performance of classical techniques for the speaker recognition task. This is the case for VQ, DTW and the discrete HMM. NNs offer an alternative solution to this problem. An enhancement to the standard MLP technique uses what is known as the radial basis function (RBF). RBF are two layered neural networks with Gaussian like functions on the first layer and linear output units. In order to eliminate the problems of long training times and bad performance scaling, a text independent RBF system (Oglesby & Mason, 1991) has been reported. The system was evaluated with 40 speakers for the task of speaker verification. This neural network model makes use of 'hand crafted' hidden units and has certain advantages such as reduced computational cost. Useful information obtained from the hand crafted units is incorporated into the model and the weights can be easily analysed. Each speaker's model uses clusters (training data are clustered using the standard vector quantization algorithm) derived from only their speech, and those derived from multi-speaker training data. Having only a single adaptive layer, the training process is said to be much faster than the MLP. Cepstral features are used in these experiments. For a 4 digit test utterance there are 210 true client scores versus 8190 impostors scores. The numbers of units used in this model are 32, 64, 128, 256 and 384. The best performance comes from an RBF network with 384 units giving 8% false rejection and 1% false acceptance.

In section 2.3.5, NNs for text dependent speaker recognition approaches are seen as excellent choices to check the authenticity of a speaker. However, there is evidence (Bennani & Gallinari, 1994) that this form of NNs suffers badly when the number of talkers is increased. Two solutions are offered to this problem. First, the idea of breaking the complexity of the NN to smaller NN architectures thus forming a modular classification systems is considered. Second, the idea of

having a predictive system to incorporate a new talker at the expense of longer test utterances than the former approaches mentioned.

As mentioned earlier the text independent task requires large amounts of data and training such a system to perform the required task can be time consuming. A possible solution is the modular system approach. (Bennani & Gallinari, 1991) for example divide the text independent task into smaller subtasks and they use a NN for each subtask. Their aim is to have the smaller network configuration necessary for the global task and therefore ending with networks which are easier to train. There are three modules developed into the system. The first module discriminates between males and females. The second and the third module are specialized in the identification of male and female speakers respectively. LPC features are used as the inputs to the three networks. It is a known fact that speech signals vary over time. To decode these signals, neural networks must be able to use appropriate representations of time. The problem of time has been addressed by the development of a Time Delay Neural Network (TDNN). The TDNN is a multi-layer feedforward NN which can be trained to recognize specific spectral structures within consecutive frames of speech. The time delay input frames account for variations in the spectral representation of speech and the weights in the initial layer will correspond to these variations. This network is trained based on the back propagation algorithm. This model has been tested on 20 speakers with the TIMIT database. The best performance of the system provided an average identification rate of 98%.

(Rudasi & Zahorian, 1991) suggested another alternative for the classification problem. Text independent talker identification is based on the binary partitioned approach. This method used $N*(N-1)/2$ binary pair classifiers for partitioning a large classification problem. The database consisted of 47 speakers taken from the TIMIT database. The cepstrum coefficients were used in this case. The large network consisted of two layer perceptrons. The time required to train a single network to perform an N-way classification is nearly proportional to the exponential of N. There were three experiments carried out for such a network with N=5, N=10 and N=15. From

the observation, increases in N tends to degrade the performance of the single layer perceptron. In fact, it was shown that the partitioned approach performed as well or even better than a single layer network. This fact suggested that more training data is required with the single large network than the partitioned network as N increases. In the approach presented, the classification scheme is well represented with large number of categories of speakers. One limitation of this approach is that the number of networks is proportional to the square of the number of speakers, which is a prohibiting factor in practice.

Hattori (Hattori, 1992) on the other hand, applied Predictive Neural Networks (PNN) to ASV. This type of network is a non-linear predictor based on the MLP. This type of network will predict non linearly the next frame from the preceding several frames (Nakamura & Akabane, 1991)(Paoloni *et al.*, 1997). This network is trained for each speaker. Hattori proposed a speaker recognition algorithm based on this type of network for text independent speaker identification task. The predictive neural network trained for a given category will have a minimum prediction error corresponding to the given input. Depending on the input, the value obtained will have wide variations and this will cause problems in deciding the acceptance or rejection of each input with a fixed threshold. Further attempts were made to improve the above approach. (Hattori, 1994) describes a prediction error normalization algorithm using the predictive neural network trained for multiple categories. This method requires the use of the prediction error obtained by a network trained for multiple categories so as to have a measurement of predictability of each input. Using the error obtained, the prediction error of an individual trained category network was normalised based on this error. If an input includes an unpredictable factor, the prediction error would increase for both networks. This factor can be cancelled by comparing the two prediction errors. In his experiments 12 speakers were used for performance measurement of the system. The system without normalization produced an EER of 41.2% and further improvement of the system performance was achieved with normalization where the EER is 2.7%. The PNN performance was also compared with other methods such as the VQ based method and the HMM method. It was shown that text independent speaker verification using predictive

neural networks requires only half the total number of parameters of the VQ and the HMM based methods. Some other methods for text independent speaker recognition, however, yielded good results. This include (Carey *et al.*, 1991)(Farrell & Mammone, 1994)(Thevenaz & Hugli, 1994)(He *et al.*, 1994)(Sheikhzadegan *et al.*, 1995) and (Artieres & Gallinari, 1995). Carey and co-workers used alpha-nets for speaker verification system; Farrell and Mammone used the vector quantization classifier and modified neural tree network (MNTN) for speaker recognition; Thevenaz and Hugli recovered some of the information ignored by VQ (the principle is to directly compare statistical distributions of samples) and applied it to speaker recognition; He, Li and Palm used the hybrid approach which combines supervised and the unsupervised learning for a speaker identification system; Sheikhzadegan and co-workers used the modular approach for speaker classification with different phoneme groups and finally, Artieres and Gallinari applied multi-state predictive neural networks for text independent speaker recognition.

Discriminative NNs models, modular classification systems, binary partitioned models and PNNs are different types of approaches to solve the many problems of speaker recognition. The NN paradigms that have been introduced, are essentially extension of multi layer neural networks. Slight variations on a single NN topology have resulted in the unique properties which best match the given task. NNs with different properties and functions can often be combined to perform a specific task. Thus, further enhancement of the speaker verification system can be obtained. Enhancements based on suitable selection of preprocessors, combination of supervised and unsupervised learning and emphasis of inter-speaker variability are carried out in this thesis and will be discussed in the following chapters.

2.3.6.2 Hidden Markov Based Methods

There are also many text independent speaker verification systems which use HMM methods. Temporal variations over a long period of time can be modelled by the Markovian transitions between states. In section 2.3.5, Rosenberg has shown that only a small difference in per-

formance was achieved between phone-like units (PLU's) and acoustic segment units (ASU's) based representation. The ASU based SV method has been tested in both text dependent and text independent mode. Rosenberg compared his work with that of (Tishby, 1991) for the ASU based system. Tishby on the other hand had expanded on (Poritz, 1982) HHM based system. In Poritz's works a five state HMM was used with all possible transitions between states allowed. This is to classify the speech segments into one of the broad phonetic categories corresponding to the HMM states. A linear predictive HMM was adopted to characterize the output probability function. In Tishby's models, the states are described as a linear combination (mixture) of autoregressive sources. The HMM model used 8 state ergodic autoregressive mixture models with 2 to 8 mixture components per state. The same database was used in Rosenberg's and Tishby's experiments in order to compare the ASU based verification experiments. With the same number of spectral vectors to represent the speakers the EER for the ASU based system is 1.1% for 7 digit long utterances compared to 2.0% for Tishby's system. The improvement of Rosenberg's system is mainly due to the introduction of more temporal detail and structure in the speaker model.

HMM methods normally trained with the maximum likelihood estimation in which model estimation is based on maximising the likelihood of the training data over all training utterances. In this approach, the technique is based on maximum likelihood estimation of a Gaussian mixture model to represent the speaker identity (Rose & Reynolds, 1990). Gaussian mixtures are robust as a parametric model and have the ability to form smooth estimates of arbitrary underlying densities. A different kind of approach called discriminative training has been suggested to solve this SV problem. There are three forms of discriminative training: maximum mutual information estimation (MMIE) (Normandin *et al.*, 1994), minimum discrimination information (Epraim & Rabiner, 1988) and minimum classification error (MCE) (Liu *et al.*, 1994). These studies show that discriminative training offers an improvement over the maximum likelihood estimation for the SV system.

Most text independent speaker recognition approaches make use of features based on long term statistics. In this approach it is impossible to model or match events at the word or sentence level such as that used in text dependent recognition. As the distribution of feature parameters in this method is normally broad or multi-modal (example male and female or different dialects), it is thus much harder to separate phonemes using this speaker model. Also, this model neglects the useful characteristics of the speaker. Consequently this model is less accurate than for text dependent speaker recognition. So in order to capture the speaker specific contents to enhance the performance of this system, a speaker adaptation technique is introduced into the system. The adaptation technique is often used in speech recognition systems to improve models trained with limited data. Experiments using this approach have been carried out (Furui, 1991)(Rosenberg *et al.*, 1990) and results indicate that intra-speaker variability which is introduced with the passage of time is more than compensated for by the robustness of the adapted model.

2.3.6.3 Dynamic Time Warping and Vector Quantization Based Methods

Yu, Mason and Oglesby compared speaker recognition performance based on three common approaches of VQ, HMM and DTW (Yu *et al.*, 1995). The emphasis of the experiments was evaluation of performance with incremental amounts of training to determine the best approach. Both text dependent and independent experiments were carried out to achieve the goal. The VQ method used the LBG algorithm (Linde *et al.*, 1980) and the DTW technique followed that used by (Furui, 1981a). The HTK toolkit was used for the continuous density HMM (CDHMM) technique (Young & Woodland, 1992). The database used was a high quality microphone in a quiet environment. This database is a subset of the BT Miller speech data with speech signals transformed into cepstral coefficients. The overall findings suggested that DTW performed better than both VQ and the CDHMM. Some speaker specific time sequence information within the speech is completely lost in VQ, however this is captured by DTW. The lack of recognition sensitivity to the number of CDHMM states suggests that the state transition probabilities do not themselves contribute to discrimination, but merely to align speech events to states. Several

suggestions were made by the authors to enhance the advantages of the different models for the speaker recognition task. Another study, (Matsui & Furui, 1992) on the other hand compared the HMM based method with VQ in relation to the robustness against utterance variations. It was found that the continuous HMM is superior to the DHMM. The continuous HMM is as robust as the VQ based method with enough training data. However, if the training data is limited the VQ based approach outperforms the continuous HMM. The study also indicates that the information on transitions between different states is ineffective for text independent speaker recognition. This means that the performance of continuous HMM is strongly correlated with the total number of mixtures irrespective of the number of states.

Recently, a newer approach to VQ modelling was proposed by (Ng *et al.*, 1995) based on discriminative training. The aim of this approach is concentrated more on the discriminative features among speakers rather than the speech features themselves. In contrast to the conventional VQ method which only considers the minimum distance to a speaker codebook, the new approach takes into account of the distances to all competing classes and all codewords in a speaker codebook. The aim of this approach is to minimize the recognition error rate. Speaker recognition experiments were carried out on a database of 200 French speakers. According to the study, EER of 0.7% was obtained for speaker verification and 0.4% for speaker identification. This new approach is found to be suitable for training with limited data and also with large population of speakers. This approach performs better than the conventional VQ algorithm.

2.3.7 Text Prompted Speaker Recognition

As has been discussed earlier in the section, text dependent speaker recognition using key words or sentences is usually more reliable than text independent systems. As argued, this approach uses specific text information which carries important speaker characteristics. An important application of fixed text can be in access control. In such situation, the speaker is required to provide the desired identification by uttering specific key words or utterances. Such a system as noted by (Furui, 1997) is normally defeated by the recording of the registered speaker's

voice uttering the key text. It would be an advantage as well as achieving high security levels if arbitrary text is generated every time the recognizer is used and the voice is only accepted when the true client speaker utters the prompted text. The vocabulary used in this system is unlimited. This has an advantage where impostors cannot know in advance the sentence that will be prompted. Another advantage to this system is that it not only accurately recognizes the speaker but it can also reject utterances that differ from the prompted text.

Currently, there are four different approaches based on HMM technology for speaker verification task. These include text independent, text dependent with predefined sentences, text dependent with randomly prompted key words and text prompted with different sentences at different sessions. In a text independent speaker verification a unique HMM captures the speaker individuality independently of the phonetic content of the input sentence. A model of the claimed speaker is matched with the input utterance and a likelihood score is obtained to be matched with a threshold in order to arrive at a decision related to the claimed speaker. On the other hand the text dependent mode asks the user to speak predefined sentences. These sentences can be the same or varied every time the recognizer is used. In the case of randomly prompted key words, it is normally referred to as text prompted (Rosenberg *et al.*, 1991). A newer approach to the text prompted speaker recognition with the specific advantages mentioned earlier is proposed by (Matsui & Furui, 1992)(Matsui & Furui, 1993). An emphasis in the design of this system is to make stable phoneme class models of each speaker using only small amounts of data. The method uses speaker specific phoneme models as basic acoustic units. The phoneme models are represented by Gaussian mixture continuous HMM or tied mixture HMM. They are made by adapting speaker independent phoneme models to each speaker's voice. With the known text of the training data, these utterances can be modelled as the concatenation of phoneme models and can be adapted through an iterative algorithm. Phoneme models that are not included in the training set are adapted based on the tied mixture HMM. During recognition, the system concatenates the phoneme models of each registered speaker to create a sentence HMM according to the prompted text. Given the input speech a likelihood score is obtained from the sentence model

so as to make a decision. If the likelihood score of both the text and speaker is high enough, the speaker will be accepted as the desired speaker. In the case of this prompted text, different sentences are uttered at different sessions, this means that the range of the likelihood score can be wide. In order to have a stable threshold, the likelihood score is normalized based on the likelihood ratio or posteriori probability. The effectiveness of the system was evaluated with the combined speaker adaptive phoneme models and a phoneme independent speaker model. The system was evaluated with 15 speakers. The database was recorded on three sessions over six months. The text prompted speaker verification system was able to reject 98.5% of speech uttered by the client speaker that differs from the key text and achieved a 100% verification rate (Matsui & Furui, 1993).

2.3.8 Speech Databases to Model Inter and Intra Speaker Variability

From the above review, there have been a variety of approaches to speech features and pattern matching techniques applied to speaker recognition systems. One of the important resources for research on speaker verification is a speech database. It is deemed necessary that a standard database be developed and that such a database should be used by the research community in speaker recognition. This will provide a reliable measure of progress in this area and further development of newer technology to replace or improve the existing one. Thus, the usefulness of a common database cannot be overemphasized and its contributions for scientific investigation are :

- Speaker verification performance benchmark.
- Investigation related to mismatch problems such as mismatch between telephone handsets and/or between fixed and mobile telephone network.
- Factors that influence the error rate such as dialect, number of training/test tokens, age groups etc.
- Comparisons of whole word and sub word units based on different algorithms.

For the design, implementation and performance evaluation of speaker recognition systems, large amounts of acoustic data are indispensable. There have been tremendous efforts in developing speech database for English by European countries as well as in North America. Similar kinds of work are carried out in Japanese as well as in the Chinese language (Ching *et al.*, 1997). In the evaluation and assessment of speaker verification, normally, the tokens used for training come from many repetitions of the client's voice and only one or a few repetitions of the impostor's voice. However, SV assessment should be with many impostors' voices. In most tasks of the SV there are two types of errors. The first error is called false rejection in which the true client is rejected by the system and the second error is false acceptance in which an impostor is accepted as the valid client. The false rejection rate of the client will largely be dependent on the intra-speaker variability which means the variability within the customer's voice. On the other hand the false acceptance error rate will depend on the similarities (inter-speaker variability) of the client voice with the impostors attempting to outperform the system. The speech databases collected are subject to intra and inter speaker variability. For telephone quality speech the variation of channel characteristics (handset types and different networks) will add to the measured intra-speaker variability. There are SV databases designed to cover the long term variations in the speaker characteristics and also databases which look into the channel variations. According to the review carried out in section 2.3.2, better performance results are achieved with clean speech data while section 2.3.3 provides evidence that the two main sources of variations from the telephone channel and the handset mismatch contributed to the increase of the error rate. Thus, keeping track of the source of variations is important when performing SV. Also, if the source of variations is isolated, this will limit the size of the database.

Collection of these databases is subjected to certain requirements which are related to speaker characteristics (sex, age , weight, height etc), regional factors and channel variation (handset and network types). It is obvious that the cost of developing such databases is prohibitive. Thus, an efficient strategy to collect, validate and distribute these databases is essential in order to promote their use in the research community. There are presently several organizations taking a

positive role towards this development. The Linguistic Data Consortium (LDC) distributes most of the American public databases. In Europe, the European Language Resources Association (ELRA) is responsible for the development of the speech databases for all the official languages of the European Union. Appendix B contain a list of available databases utilized in most of the experiments for the SV task.

2.4 Speech Recognition Technology

Another aspect of speech technology is in the area of automatic speech recognition. Studies on the speech recognition problem have been in progress for some time now. Early studies were devoted to small vocabulary isolated word systems. Further studies were carried out on large databases for continuous speech and more recently hybrid systems combining statistical models and neural networks have been developed.

2.4.1 Neural Networks for Speech Processing

Lippmann's article "Review of Neural Networks for Speech Recognition"(Lippmann, 1989) is one of the most widely referenced papers for researchers working in speech technology. Many advances have been made with more recent approaches or techniques. The interested reader can refer to (Waibel & Lee, 1990) for further developments and insights on speech recognition. Below is a brief summary regarding application of NNs for speech recognition.

2.4.1.1 Static classification

(i) Multilayer neural networks:- Multilayer perceptrons have been applied to speech problems by various researchers such as Peeling and Moore, Lippmann and Gold, Kammerer and Kupper, Huang and Lippmann, Elman and Zipser and Kohonen. The databases used were sets of words and digits. As an example, experiments carried out by Elman and Zipser (Lippmann, 1989)

show that hidden nodes often become feature detectors. In Lippmann and Gold (Lippmann, 1989) neural networks with a single layer provide poor performance on digit recognition based on the Texas Instruments database. Experiments carried out by Kammerer and Kupper (Lippmann, 1989) and Peeling and Moore (Peeling & Moore, 1988b) and Burr (Burr, 1988) show good performance on small vocabulary word recognition. The outstanding results obtained from small vocabulary words and digits suggest that a real time system can be implemented using analog neural networks with VLSI processing.

(ii) Hierarchical neural networks:- Hierarchical neural networks when applied to classify speech patterns have the advantage of rapid training and the ability to combine supervised and unsupervised learning. This net when compared with a two layer perceptron needs less training data. Work carried out by Huang and Lippmann on vowel classification has the lower feature map trained in unsupervised mode (as a vector quantizer) and the upper map trained with supervised learning. Kohonen and co-workers compared a neural network classifier called learning vector quantizer (LVQ) to Bayesian and kNN classifiers. The LVQ classifier has the lowest error rate compared with the other classifiers. In another set of experiments on the same problem Kohonen showed that Boltzmann machines provide near optimal performance followed by LVQ classifiers and multilayer perceptrons (Lippmann, 1989).

2.4.1.2 Dynamic Classification

(i) Time-Delay Neural Networks (TDNN):- Researchers working with TDNN have reported promising results in comparison with HMM. Waibel performed experiments using 2000 voiced stops spoken by three talkers. The neural networks provided an error rate of 1.5% compared with an error rate of 6.5% by a discrete HMM recognizer. However, the training for the neural networks took several days. More work by Waibel showed how small networks can be scaled up to solve large classification problems with reduced training time (Waibel *et al.*, 1989).

Lang and Hinton (Lippmann, 1989) devised a training technique called multi-resolution training. This involved training networks with smaller number of hidden nodes, splitting weight values to hidden nodes to create larger desired networks and then retraining the larger networks. In experiments carried out on a data base of confusable subset of the alphabet 'B','D','E' and 'V', the multi-resolution neural networks provided an error rate of 8.6%.

Unnikrishnan, Hopfield, and Tank (Lippmann, 1989) obtained a low error rate on the digit recognition problem. Other work by Gold on large speech databases of allophones and words showed no improvement in performance over the existing HMM recognizer (Lippmann, 1989).

(ii) Hierarchical networks:- McDermott and Katagiri (Lippmann, 1989) used Kohonen's LVQ classifier with the same database as Waibel. This network trained faster but required more computation and memory. There is not much difference in performance from Waibel's results.

(iii) Networks with recurrent connections. Approaches to neural networks with recurrent connections are carried out by Anderson, Robins (Lippmann, 1989) and Prager and others (Pager *et al.*, 1986). Some of these experiments used Boltzmann machines. These networks provided good performance on a small problem but with excessive training time.

Detailed descriptions of the work mentioned above regarding the types of networks, data base and performance evaluation can be found in (Lippmann, 1989). In his paper Lippmann also discussed research work integrating neural networks and conventional HMM systems. It is important to note that most of the state of art speech recognition systems based on HMM models make use of context-dependent phonetic units such as triphones while NN approaches are still restricted to phoneme models (Bourlard, 1991).

2.4.2 HMMs for Speech Recognition

HMM/VQ (Rabiner *et al.*, 1983) has been compared with the traditional classifier LPC/DTW and the result has shown a slight decrease in performance relative to the latter model. However the HMM/VQ model involves less storage and less computational complexity. The overall performance for an isolated digits system has a 96.5% accuracy based on a 100 talker test set.

Rigoll (Rigoll, 1989) managed to reduce the training time of the IBM speech recognition system to 5 minutes instead of the usual 20 minutes. However, the average recognition rate dropped by 1.2%. Further work (Rigoll, 1990) by Rigoll adapts the structure of the Markov models used in the IBM speech recognition system to the current user. There is an increase of 1.1% recognition performance for a speaker who has already trained the system.

Several analytical techniques have been improved and developed for estimating HMM probabilities. These techniques enabled the HMM to become more computationally efficient so it can be applied to many large vocabulary, speaker independent systems (Lee, 1988)(Rosenberg *et al.*, 1991)(Huang & Jack, 1988). Motivated by the success, implementations of the proven HMM algorithm on continuous speech have been developed. The availability of large speech databases has made this task easier for speech researchers and has provides a benchmark for HMM performance. Three important databases in English available today are the TIMIT Acoustic Phonetic, DARPA Resource Management and the TI/NBS connected digit. Table 2.1 shows data for reported HMM systems using large databases for speech recognition. The Tangora system is a speaker dependent isolated utterance speech recognition system (IBM) that recognises 5000 words to 20000 words. The SPHINX system is a speaker independent continuous speech recognizer based on triphone acoustic models. AT&T focus their development on digit recognition. Texas Instruments perform the same task using a technique called phonetic discrimination. Detailed descriptions of HMM speech recognition systems for large vocabulary described above can be found in Table 2.1 (Picone, 1990).

SYSTEM	ALGORITHM	RECOGNITION ERROR RATES	
TANGORA- IBM Isolated Words Speaker Dependent	VQ and Discrete HMM	5000 words 2.9%	20,000 words 5.4%
SPHINX-CMU Continuous Speech Speaker Independent	VQ and Discrete HMM Multiple codebook approach	No grammar 50.4%-18.1%	word pair 16.2%-3.8%
AT&T Digit Recognition Speaker Dependent	CDHMM	Mixture (*) 9.2% to 4.2%	Models (**) 4.35% to 3.01%
TI Digit recognition Speaker Dependent	CDHMM	Covariance (***) 3.5% to 1.5%	

* Number of mixture components (1,3,5,7 and 9). Each state is a mixture of Gaussian distributions. Digit recognition error for the case of 10 states per model with one model per digit.

** Number of models per digit (1,2,3,4,5 and 6).
Digit recognition error for the case of 10 states per model with 9 mixture components per state.

*** Pooled covariance to confusion discriminants.

Table 2.1: Speech Recognition Systems for Large Vocabulary

2.5 Conclusion

There are four main areas in speech technology- speech recognition, speaker recognition, speech synthesis and speech coding. Briefly, the goal of speech synthesis is to develop a system which can accept as input of any text and convert it to natural sounding speech. Speech coding concerns itself with the redundancy in the speech signal in order to reduce the number of bits required to represent it. The ultimate goal of speech recognition is to produce a system which can recognize to a high accuracy continuous speech from any speaker of a given language. Speaker recognition addresses the problem of identifying an unknown voice as well as verifying the claimed identity. The focus of this chapter has been on speaker recognition, particularly neural network speaker verification.

Table 2.2 summarizes the results presented for ASR based on traditional methods and connectionist methods. Typical approaches to speaker recognition include VQ, DTW, HMM and NN based text dependent recognition, text independent recognition, and the current trend of HMM, a text prompted recognition method.

Conventional approaches have achieved excellent results in speaker recognition. The primary concern for this task has also been to minimize false rejection and false acceptance rates. One important factor in speaker recognition is the selection of features to discriminate among the population of speakers. Previous reviews show that parameters such as pitch, formant and predictor coefficients even though sucessfully used do not guarantee to be unique to the client speaker. Thus, NN offers a way to extract features for the speaker recognition task. In addition, NN are used for classification or clustering of the speakers' features. These methods like MLP, RBF and TDNN have the potential to provide excellent performace. Also, NN when applied to a large problem can cause performance deterioration. This can be seen in most of the studies carried out using the back propagation algorithm. Several strategies are examined to apply such networks to large scale problems. One strategy is based on modularization. This can be in the form of breaking a large task into much simpler tasks that can be solved with a much smaller

network configuration. The strategy of normalizing the prediction error has brought about a system design that involves fewer parameters than conventional methods. In the field of speaker recognition, NNs have played an important role in the development of this technology.

Study		Type	Approach	Subject	Feature vectors	Recognition and EER Rate (%)	Comments
Soong	1985	ASI	Speaker based VQ	100	LPC	98	telephone speech
Buck	1985	ASI ASV	Speaker based VQ	16 + 111	-	FR-0.8 FA-1.8	16 client 111 impostor
Gish	1985	ASI	Gaussian pdf classifier	10	CC	-	Shows effect of telephone channel on ASI performance
Gish	1986	ASI	Probabilistic channel model Channel invariant model Modified Gaussian model	20	CC	-	Developed methods dealing with problems of telephone channels
Rosenberg	1986	ASV	Speaker based VQ- DTW	100	LPC	TI EER-2.2 TD EER-0.3	Talker recognition system can be easily extended from TI to TD
Federico	1987	ASI ASV	Statistical Algorithm	5	CC	-	telephone speech
Oglesby	1988	ASI	MLP	-	LPC	-	Results not given
Xul	1989	ASI	Speaker based VQ	-	LPC PLP	-	Optimization of features for VQ codebook
Naik	1989	ASV	Speaker based DTW-HMM	20	LPC	DTW EER-6.2 HMM EER-2.3	Better performance on the HMM(discriminative model)

key: * FR-false rejection, FA- false acceptance, EER -equal error rate
* LPC -linear predictive coding, ASI- automatic speaker identification, ASV- automatic speaker verification
* CC- cepstral coefficients, DTW- dynamic time warping, HMM -hidden Markov model, BP - back propagation
* FFT- fast fourier transform ,TI -text independent, TD- text dependent, PLP- perceptually weighted feature

Table 2.2: Experiments and Results in Automated Speaker Recognition

Study		Type	Approach	Subject	Feature vectors	Recognition and EER Rate (%)	Comments
Xu	1989	ASI	Speaker based VQ	-	LPC PLP	-	Advantages of PLP features over LPC demonstrated
Jou	1990	ASV	MLP	20	LPC	Inside test (98.5) Outside test (98.3)	Speaker dependent features are not easily fetched
Yin	1990	ASI	MLP	20	LPC	TI-95.2 TD-98.5	Equal performance to VQ based system
Oglesby	1990	ASI	MLP	10	LPC	92	Equal performance to VQ based system
Bennani	1990	ASI ASV	LVQ	10	MFCC LPC	97	Connectionist model can perform ASV
Rosenberg	1990	ASV	HMM	100	LPC	-	Good performance obtained using HMM sub word units
Logan	1990	ASV	VQ/HMM	5	LPC	-	Hardware implementation
Careyl	1991	ASV	HMM	50	MFCC	FA-2.6 FR-6.4	Trained with the Baum Welch algorithm and partial derivatives of BP telephone speech
Nakamura	1991	ASV SR	NPM LMQ	10	FBC	-	VQ based system performed better than NPM but LMQ has better results
Oglesby	1991	ASI ASV	RBF	40	CC	CR-92	Better performance than the MLP and the VQ based systems. telephone speech

Key : * MFCC- mel frequency cepstral coefficients, FBF- fourier Bessel function, SR- speech recognition
* NPM- neural predictive modeling , LMQ- learning matrix quantization, FBC- Filter bank coefficients
* RBF- radial basis function, MLP- multi layer perceptron, LVQ- learning vector quantization

Table 2.2: Experiments and Results in Automated Speaker Recognition (continue)

Study		Type	Approach	Subject	Feature vectors	Recognition and EER Rate (%)	Comments
Gaganelis	1991	ASV	FBF	45	CC	-	telephone speech
Bennani	1991	ASI	TDNN	20	LPC	98	TI
Hunt	1991	ASV	HMM	-	-	-	Commercial system for ASV and SR telephone speech
Matsui	1992	ASI	DHMM CHMM	36	CC	-	CHMM has equal performance to VQ based method
Bennani	1993	ASI	STDNN	-	CC	-	Better performance compared with MARM
Forsyth	1993	ASV	DHMM	10	CC	EER-4.0	telephone speech 10 training token models
Forsyth	1993	ASV	SCHMM	10	CC	EER-2.0	SCHMM performs better than DHMM
Farrell	1994	ASV ASI	VQ/MNTN	100	CC	-	Supervised and unsupervised classification
Hattori	1994	ASV	PNN	12	FFT	-	Overall performance poor but used fewer parameters compared with VQ and HMM methods
Fredrickson	1994	ASI	RBF	12	MFCC	-	Better performance than the MLP
Tsoi	1994	ASV	RNN	34	DDC	MLP-94 RNN-92.5	MLP has better performance than RNN
Anderson	1994	ASI	SOFM	-	AIM PAM LPC	-	Different auditory model representation to perform ASI

Key : * TDNN- time delay neural networks, STDNN- shift TDNN, DHMM- discrete HMM, CHMM- continuous HMM
* MARM- multivariate auto-regressive models , SCHMM- semi-continuous HMM, MNTN- modified neural tree net
* PNN- predictive neural networks, SOFM- self organizing feature maps, AIM- auditory image model
* RNN- recurrent neural networks, DDC- delta delta cepstra, PAM- Patterson & Payton's auditory model

Table 2.2: Experiments and Results in Automated Speaker Recognition (continued)

Study		Type	Approach	Subject	Feature vectors	Recognition and EER Rate (%)	Comments
Falavigna	1995	ASV	CHMM	138	MFCC	EER-1.3	Text Prompted
Bonifas	1995	ASV	DTW/VQ	10	LSF	FA -2.2	Threshold adaptation
Yu	1995	ASI ASV	CHMM DTW/VQ	-	CC	-	Best result DTW
Gong	1995	ASV	Bayes Statistical	-	MFCC	EER-0.65	Strategies for Thresholds
Hussain	1995	ASV	HMM/MLP	11+ 83	CC	EER-1.04	Suitable preprocessor For MLP. 11 client and 83 impostor
Wagner	1996	ASV	VQ/HMM	-	MFCC	EER-4.0	Combination of speech and speaker recognition
Carey	1996	ASI	HMM	-	CC	-	Handset variations
Vuuren	1996	ASI ASV	VQ/GMM	20	PLP	-	Handset and Channel Variations
Hoge	1997	ASI ASV	SpeechDat	120	-	-	European speech database
Imperl	1997	ASI ASV	Average distance	50	Harmonic Features	-	Harmonic features better than LPCC
James	1997	ASV	YOHO CAVE	138	-	-	American and European Speech database
Paoloni	1997	ASV	PNN	12	MFCC	EER-0.5	Combined approach of PNN and statistical
Furui	1997	ASI ASV	Speaker Recognition	-	-	-	Recent review

Key : * TDNN- time delay neural networks, STDNN- shift TDNN, DHMM- discrete HMM, CHMM- continuous HMM
* MARM- multivariate auto-regressive models , SCHMM- semi-continuous HMM, MNTN- modified neural tree net
* PNN- predictive neural networks, SOFM- self organizing feature maps, AIM- auditory image model
* RNN- recurrent neural networks, GMM - Gaussian Mixture Modelling, EER- Equal Error Rate

Table 2.2: Experiments and Results in Automated Speaker Recognition (continued)

Chapter 3

SUGGESTIONS FOR FURTHER DEVELOPMENT OF SPEAKER VERIFICATION SYSTEMS

3.1 Introduction

In chapter 2, the relation of SV to a variety of pattern matching approaches was discussed. In pattern recognition, speaker identification and verification can be challenging tasks. The techniques, evaluations and implementation of the speaker recognition task are reviewed with special attention given to SV. The two main speech pattern recognition problems reviewed are speech and speaker recognition and most attention has been given to speaker recognition. Speech and speaker recognition have received a great deal of attention among speech researchers. It would not be possible to make an exhaustive survey covering the two main topics. Rather, attempts are made here to show how the theory and techniques used in speech recognition are common to speaker recognition. Overwhelmed by the success of speech recognition systems, researchers have applied the same techniques to speaker recognition systems. Since the field of speaker recognition is still young, the possibility of using techniques in speech recognition has not been fully pursued. This chapter contains a general summary of potential enhancements in the field of speaker verification, to provide a background for the novel work carried out in this thesis. The techniques presented in section 3.2 are procedures to normalize, or represent a variable length set of features with a fixed number of neural network inputs. Section 3.3 describes the database

used to evaluate the neural network speaker verification systems. Section 3.4, describes the effects of long term spectral variations caused by the speaker as well as channel characteristics. Finally, the overall design of an automatic speaker verification system is described.

3.2 Temporal Alignment of Utterances

Variation in the temporal word length poses a problem for neural networks with a fixed length input layer. There are several methods that can transform the segmented word to a fixed number of parameters. These include linear time normalization, trace segmentation and non-linear time normalization. Speech can be considered as dynamic and complex in both spectral and time domains. At the word level temporal structure exists in the form of phonemes, which are essential for speech perception. Two speech utterances with the same text and spoken by the same speaker do not have the same temporal alignment of speech. This can be attributed to the differences in the speaking rate. By properly setting the beginning and the end points of the two utterances correct aligning of the two speech signals can be achieved. The important characteristics of the speech are obtained in the time varying nature of the speech rather than the steady state information. (Peeling & Moore, 1988b)(Peeling & Moore, 1988a)(Oglesby & Mason, 1990) (Bourlard, 1991) have transformed the sequential inputs into static like patterns and obtained good results.

Word length variation can be addressed by dynamic time warping (DTW) which can select the feature vectors for a fixed length neural network input. The technique of dynamic time warping (DTW) is useful to time normalise the non-linear speech signals. The use of DTW assumes that the input speech signals as well as the reference speech have been transformed into temporally sequential of frames. Each frame of the input utterance is compared using a simple Euclidean distance measure with several frames of the reference utterance. The warping process between these two speech signals is confined by the local and the global constraints.

The frame which results in the lowest accumulated distance is matched to the input frame and the process continues. The result of this process is a time aligned utterance pair. One example is the dynamic programming neural network (DNN) proposed by Sakoe and others (Sakoe *et al.*, 1989) which combines the dynamic programming (DP) and the MLP. This approach has the important features of DP and neural network.

Another form of normalization is trace segmentation (Demichelis *et al.*, 1989) (Zhu & Fellbaum, 1990). Trace segmentation uses the decimation or interpolation technique. Trace segmentation which uses the decimation technique (Morgan & Scofield, 1993) computes the trace of a function in a feature space. The speech is represented as a path in some feature space. Different utterances of the same word should follow the same path and the time taken to traverse the path should differ. An advantage with trace segmentation is that rapidly changing events are retained while the steady state events are discarded. Another form of trace segmentation uses linear interpolation to form new frame representations from the existing frames and this can also result in words having a fixed length (Gauvain *et al.*, 83).

The MLP can be used as a postprocessor to make classification decisions for input utterances time aligned with linear time normalization (LTN) of the speech signals. Of all the techniques presented, LTN is a simple procedure to normalize a variable length of speech utterance into a fixed number of parameters. To properly present the temporal structure of speech in the static classification of neural network, the LTN technique has been applied in the work carried out in this thesis. The problem with LTN is that phonetic events (especially short time events) such as the plosive can be discarded during the process or insertion of feature vectors which may alter the relative duration of the plosive. This is more important than the steady state information in the vowel. Care must be taken in selecting the proper LTN values which preserve the information of the speech signals. It is of interest to study the effects of different values of LTN (with respect to information being discarded or inserted) to the speaker verification performance. Experiments to

compare these different values of LTN have been performed, and the details of these experiments and their results are given in chapter 5. A common disadvantage of the linear time normalization and trace segmentation is that both methods can discard or insert information.

These are three methods which map the temporal sequence into a fixed length that can be used with the static neural network. Other techniques still exist to handle the dynamic properties of the speech signals. As the front end processor the hidden Markov model can be used to provide the local distance scores for the MLP. Recurrent neural networks have also been proposed. By including a short time delay in the input and the hidden layers of MLP, feedback loops can be added providing dynamic and implicit memory. Researchers (Bourlard, 1991)(Shrimpton & Watson, 1992) have used such networks where the speech signal enters the network sequentially. In general these approaches are more difficult to train and analyze compared to the static MLP but are more appropriate if connected words are used.

3.3 Availability of Training Data

There are several classes of security device such as keys, personal identification number (PIN), smartcard and biometric based system such as speaker verification. Security is a critical issue to many business such as banks. An automatic service with a PIN is not secure enough to allow callers the chance of carrying out bank transactions. The technology of speaker verification offers a fast and secure automated scheme. Speech data are required in order to train the system and this can be inconvenient and time consuming to the customer. This will relate directly to customer acceptance of the system. As will be discussed in more detail later in the section, neural networks form the foundation for the SV task in this thesis. For these networks, it would be an advantage to use as much data as possible. More training data will almost certainly decrease the error rate. The main reason for this is that the more training data that is available, the more parameters that can be trained and consequently develop a more detailed model. Besides the

choice of the algorithm for verification a decision has to be made concerning the size of the training data. A very small size data set will result in a very difficult verification task. The objective of this thesis is to improve the performance of the NN SV system under the constraint of limited enrolment data (5 tokens of each digit). A training session of 3 to 5 minutes consisting of 5 repetitions of each digit seems reasonable for the above task.

In order to compare the performance of the SV system developed in this thesis with the traditional techniques of HMM by (Forsyth, 1995), the same database was used. The BRENT database contains data for speakers recorded with different calls and from different locations with different types of handsets. This data was collected from native British English speakers throughout the United Kingdom from 120 male and female speakers.

3.4 Long Term Effects on Speaker Verification

Speaker recognition performance is correlated with the variation in signal characteristics from one session to another. These variations come from noise, telephone channels, handset types and even from the individual speaker's voice. As for the speaker, for each individual there are long term effects on the voice. Voice samples recorded in the same session are more highly correlated then voice samples recorded in different sessions. Parameter normalization techniques are an efficient way to tackle the long term spectral variations caused by the speaker as well as channel characteristics. Attempts to make speaker recognition more robust to adverse conditions by enhancement or modification to various forms of cepstral processing were discused earlier in chapter 2, section 2.3.3.

It is also important to apply adaptation techniques to the reference model as well as its verification threshold to maintain high accuracy over a long period of time. Adaptation techniques have also been shown to enhance the performance of speech and speaker recognition systems.

They have been shown to be effective when systems are faced with limited enrolment data or to convert speaker independent models to speaker dependent models. This is because the speech signal can vary significantly over a short or long term as well as differ systematically in some respect from the training speech. For example, hidden Markov models have been used in speaker adaptation. The current trend in speech recognition research is to further reduce training data for new users of a large vocabulary by making use of a speaker adaptation technique. Work in this area includes research to reduce the time necessary for enrolment of a new speaker and also the aspect concerning the adaptation of the speech recognition system. The algorithms for speaker adaptation are concerned with modification of the Markov model parameters and also adapting the structure of the hidden Markov model. (Lee & Gauvain, 1993) performed speaker adaptation through maximum a *posteriori*(MAP) estimation of hidden Markov model parameters. The study shows that when compared with speaker dependent training, speaker adaptation achieved an equal or better performance with the same amount of training/adaptation data. Others like (Furui, 1989)(McInnes, 1988)(Rosenberg *et al.*, 1992) have also applied adaptation techniques to improve the recognition performance. In chapter 6 of this thesis it is explained why adaptation of the threshold as well as the client barcode will be vital to the long term performance of the NN SV system.

3.5 Speaker Verification System Design

3.5.1 Preprocessor Selection for Better Classifier Generalization

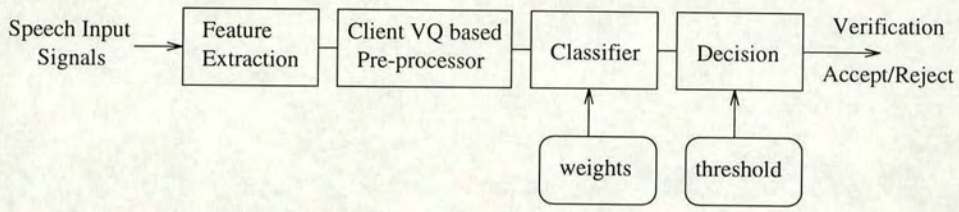
A brief description of the verification system is presented. For each word spoken, the feature extraction stage will convert the raw speech into the desired features referred to as the training or test token. One of the most important issues of designing a NN for speaker verification is a way to handle the dynamic properties of the speech signal. Of the many approaches, the simplest solution to this problem is to apply a static NN with the procedure described in section 3.2.

Another important issue is the complexity of the network system in relationship to training time and limited data. Selection of a network which minimizes the input will reduce the number of patterns required for training. There are a variety of means by which the preprocessing stage can be structured to solve the specific SV task. The design of the current study for the preprocessing stage can be in any of four basic forms. These are illustrated in Figure 3.1 and Figure 3.2. The first of these only makes use of useful information from the preprocessor so that the highly redundant speech data are reduced. The second uses multiple codebooks for data efficiency to enable direct modelling of the differences between the client and the impostor speakers. The third model emphasizes the similarity and dissimilarity between the client and the impostor speaker. Finally, there is a preprocessing model that handles the statistical variation in the spectral features. The first and the second model are discussed in chapter 5 while the discussion of the third model is in chapter 6. The fourth model needs further elaboration and will be discussed further in the next section.

The next stage is to use a classifier with the preprocessed data mentioned above. Each client speaker has a network that is trained to respond to the client utterances. Including data from the general population will enhance the ability of the classifier to discriminate between the authorized client and the impostors. In order to verify the claimed identity of an unknown speaker, in most applications the speaker is requested to repeat a series of isolated words chosen

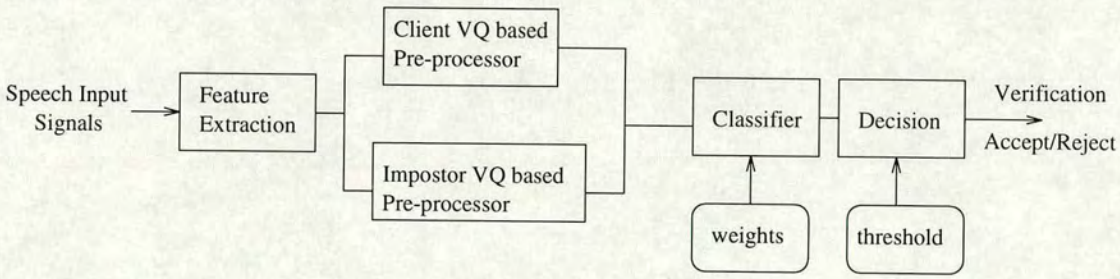
at random from the claimed identity's speaker subvocabulary. After each word, a sequential decision up to N words is determined. As the number of words increases the reliability of the sequential decision score decreases. The sequential decision score is compared with the desired threshold in order to make the verification decision of accepting or rejecting the claimed identity of that person. The use of the sequential decision gathers the information in such a way that it can accumulate the score until an accurate decision can be made. There is a trade off between the amount of information obtained and the performance of the system. If the client is accepted as the claimed speaker, the system can be made to update the reference data of the newly spoken word through the process of adaptation mentioned in the last section. This updating is necessary to keep track of the changes of the client speaker's voice as well as to maintain the performance of the system. The system can be made to update only when the comparison score is below the prefixed threshold. Different performance measures are used to evaluate the system as can be seen in chapter 7.

MODEL 1:



Elimination of redundant speech through a process of vector quantization.

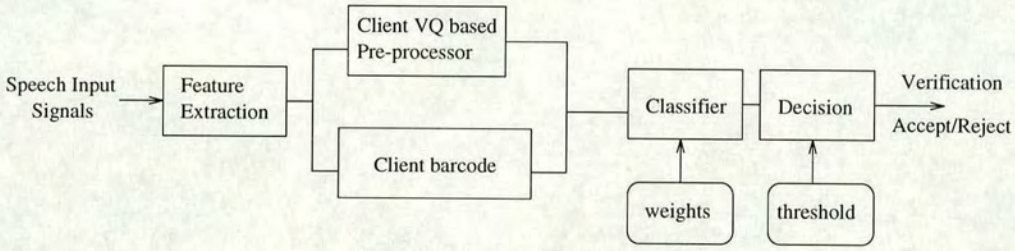
MODEL 2 :



Multiple codebook approach for data efficiency.

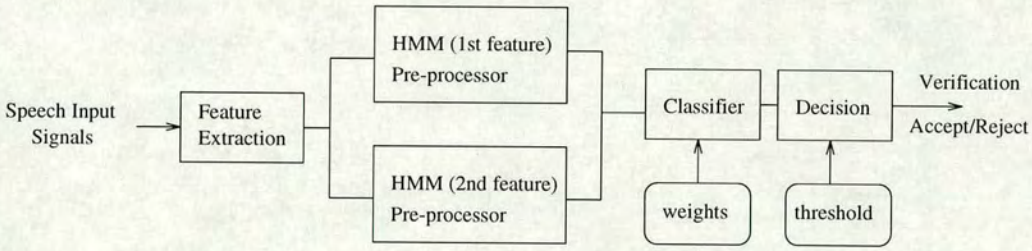
Figure 3.1: Two Preprocessor for the Speaker Verification Task.

MODEL 3 :



Additional features with further emphasis on similarity/dissimilarity between the client and the rest of the population.

MODEL 4



An approach to efficiently model the statistical variation in spectral feaures.

Figure 3.2: Further Preprocessors for the Speaker Verification Task.

3.5.2 Hybrid Approach to Speaker Verification

Several companies such as IBM, AT&T and Texas Instruments have developed large vocabulary speech recognition systems that represent advances in the state of art technology. However, HMM has not reached its full potential and researchers are concentrating their efforts towards detailed speech models, large databases and improved training algorithms. HMM techniques applied to ASR problems have yielded an improvements in recognition accuracy. Neural networks on the other hand offer a new algorithmic approach to problems in speech recognition. In many of the findings, neural networks work well with limited vocabularies but the approach has yet to prove its performance for large vocabulary and continuous speech. Neural networks have difficulty in handling the time sequential nature of speech while hidden Markov models have poor discriminative property. These two approaches have shown promising results but they still have weaknesses and limitations. Table 3.1 shows some of the disadvantages of HMM and NN.

Combining these two techniques has brought about a significant improvement in performance for speech recognition systems. Even though there are many advances and promising results in speaker recognition there are still many problems for which good solutions need to be found. The interested reader can refer to a review of speaker recognition technology by (Furui, 1994) (Furui, 1997). These papers provide an overview of recent advances in the domain of speaker recognition with special emphasis on the improvement made possible over conventional systems. On the other hand, (Bennani & Gallinari, 1994) focus on speaker recognition systems based on connectionist models.

Several approaches have been proposed to combine HMMs and NNs. These approaches try to take advantages of the two systems in order to improve the performance of speech recognition and speaker recognition systems. The hybrid approach can be divided into several approaches: studies on the combined approach of neural networks with DTW (Lippmann, 1989); connectionist models used as a probability estimator (Bourlard, 1991)(Renals & Morgan, 1992)(Bengio

HMM models	NN models
Poor discrimination due to training algorithm which maximizes likelihoods instead of a posteriori probabilities.	High performance obtained only with small data base isolated words.
Different HMM topological structure leads to different speech frame segmentation.	Inability to deal easily with time sequential nature of speech.
Assumption that the state sequences are first order Markov chains.	MLP's used as a probability estimator have larger networks:- training can be complex and inappropriate for current digital computers.
	MLP's used as a probability estimator is restricted to phonetic HMM.
	For continuous words and isolated units: there is no principled method for selecting the target function and this does not appear in HMM approach.

Table 3.1: Disadvantages of HMM and NN Models

et al., 1992); hidden Markov models within a connectionist framework (Young, 1991); and connectionist models as a vector quantizer for HMM parameters (Cerf & Compernolle, 1993).

A hybrid system proposed in this thesis is the combination of two different models. In the first stage the hidden Markov model consists of a parametric production model for each particular speech segment. This model depends on some assumptions on the statistical independence of the features and pattern distribution such as the Gaussian probability density function. Two different feature sets are used to train the HMM resulting in two output probabilities. A connectionist architecture is applied in the next stage for the classification task. The objective of the proposed hybrid experiments was to evaluate the ability of the preprocessing HMM to extract an improved representation of the input data for classification. The multiple feature sets used provided multiple information to make the verification decision. The usefulness of combining these features depends on the independence of the speaker discriminative information they contain. If the two different scores are sufficiently uncorrelated then it is likely that the combination of the scores will produce a more robust probability than either score alone (Forsyth, 1995). When compared

to the input data representation in section 3.5.1, the new feature space has certain advantages:

- better generalization with data outside the training set
- data compression, the new feature space with lower dimensionality allows the classifier to be used with a smaller network configuration and reduced training time.

In fact, the hybrid approach shows significant improvement in EER compared to the single feature set of cepstra and delta cepstra respectively. Detailed experiments with the hybrid approach are discussed in chapter 4.

Chapter 4

HYBRID APPROACH TO SPEAKER VERIFICATION

4.1 Introduction

Neural network models (NNMs) are well known as powerful learning tools in a variety of tasks related to speech and speaker recognition as discussed in detail in chapter 2. Lippmann (Lippmann, 1989) reported research work that combines conventional HMM and Dynamic Time Warping (DTW) speech recognition algorithms with neural networks. One of the most important developments in recent years for speech recognition has been the combined approach of neural networks with HMM. Along similar lines, researchers have attempted to create hybrid systems for speaker recognition. A number of recent studies have shown that combining neural network discriminative abilities with the automatic scoring and training algorithms used in HMM can result in a system that is better than either MLP's or HMM's (Naik, 1994)(Cerf & Compernolle, 1993)(Bourlard *et al.*, 1993).

4.1.1 Hybrid Approaches to Speaker and Speech Recognition

4.1.1.1 Multilayer Perceptron, Dynamic Time Warping and Vector Quantization

Lippmann (Lippmann, 1989) published a list of researchers working in these areas. An experiment carried out by (Boulard & Wellekens, 1989) shows MLP integrated into a DTW continuous

speech recognizer. The performance is slightly better than the discrete HMM. Other researchers who have followed the work of Bourlard and Wellekens are Sakoe and Iso. They used delays in the network, however no results were presented for this approach (Sakoe & Iso, 1987). Lippmann and Gold define a neural network architecture called "Viterbi Net" that can implement a DTW decoder. The Viterbi net was evaluated using 4000 word tokens from the 9 talker, 35 word Lincoln Stress Style speech data base. This network has the same capability performance on a large database as has a robust HMM isolated word recognizers (Lippmann & Gold, 1987).

Another robust system was proposed based on both DTW and VQ connected in series (Bonifas *et al.*, 1995). The line spectrum frequencies (LSF) coefficients were claimed to be able to handle each speaker's characteristics and could also take into account the intrinsic variation of the human voice due to stress or cold. The text dependent speaker verification system has a two stage recognition process. The first stage is a recognition process which eliminates mainly casual impostors. The second stage of the recognition process is capable of separating the client speaker and the dedicated impostors. The aligning process between two speech signals is the DTW. The system is designed to have high discrimination rate but also to use as little memory as possible. This could be achieved when the reference is stored in terms of the codebook index. The speech signals used were 2 s long utterances. The database consisted of three sentences repeated by 10 speakers over a period of three weeks. The system was also compared to a SV based only on the VQ. The results show that the use of DTW and VQ connected in series performs well for the text dependent speaker verification task. The false acceptance rate of 2.2% was obtained with the use of dedicated impostors and 0.085% with the casual impostors.

4.1.1.2 Multilayer Perceptron as the Probability Estimator

In a study carried out by Bourlard and Wellekens (Bourlard, 1991), they made a conclusion necessary to link the connectionist approach and HMM. This conclusion has been proven exper-

imentally. According to them the outputs of the MLP approximate local *a posteriori* probabilities which are known to lead to optimal classification; they are discriminant by nature and minimize classification error. They have also pointed out that the techniques, if applied to triphones, would result in an output layer with thousands of units and many millions of parameters to train. However, suggestions have been made to device estimates of probabilities with MLP's and to extend the approach to triphone models. Table 4.1 represents some of the rescent studies in hybrid NN and HMM systems. In one study made by Rigoll (Rigoll, 1992) a 26% error rate reduction was achieved if an HMM is combined with neural networks. The approach taken by Rigoll is to consider information theory principles for the training of the neural network. This results in an unsupervised learning method for the multilayer network.

In the above discussion of HMM the activation value of each output node of the network corresponds to P(observation(t)\state(i)), the probability of observing a set of acoustic parameter values at time t condition on the state i of the HMM. In the work carried out by Renals and Morgan (Renals & Morgan, 1992) MLP's were used as the probability estimator. Issues in the implementation of the connectionist model and HMM are discussed in this paper. The inputs to the neural networks made use of the DARPA database. There were 1000 hidden units and 69 output classes, giving a total of 300,000 weights. Training time took about 1-2 days using a ring array processor of TMS320C30 DSP chips. The discrete HMM system gave a word error of 11% using the Resource Management word-pair grammar (perplexity 60). When the output *a posterior* probabilities of MLP's were used the word recognition error improved to 6.2%.

Bourlard (Bourlard *et al.*, 1993) obtained high recognition scores on a task dependent (HIM) database[1] with 16.2% error rate reduction compared with the discrete HMM system. Even with

[1] HIM- contained 53 words relative to the targeted speaker. The database has been recorded in real telephone condition. It contains a lot of small and/or very confusable words (e.g. no/go, switch off/switch on).

Study	Approach	Data	Comments
SPEECH RECOGNITION			
Rigoll(1992)	MLP , HMM	Phonemes,ATR	Unsupervised learning algorithm for MLP Information theory principles Average error reduction is 25% with HMM/NN model.
Renals (1992)	MLP , HMM	DARPA	MLP's as probability estimator Continuous speaker independent system.
Bourlard (1993)	MLP ,HMM	DARPA	MLP's as probability estimator Speaker independent isolated words.
Lippmann (1993)	HMM ,RBF	NIST Road Rally	Word spotting.
Bengio (1992)	MLP , HMM	TIMIT	MLP's as probability estimator Speaker Independent Proposed an algorithm for global optimization. and HMM trained separately 81% accuracy using ANNs 86% using the hybrid system.
Cerf (1993)	MLP , HMM	Phonemes	MLP's as labelers for HMM Multiple-MLP approach Fewer HMM parameters Equal performance to the discrete HMM system.
SPEAKER VERIFICATION			
Naik (1994)	HMM ,MLP	Telephone speech Timit database	MLP's as probability estimator 20% improvement over DHMM.
Kilmartin (1992)	MLP,RBF	Clean speech	Hybrid approach performs better than the MLP SV system.
Bonifas (1995)	DTW,VQ	Telephone speech	Hybrid approach performs better than the VQ SV system. Robust features to handle the intrinsic variation of human voice.
Carey (1991)	HMM ,MLP	Telephone speech	A connectionist implementation of HMM HMM with discriminative training ability.

Table 4.1: Summary of on-going Research in Hybrid HMM, DTW And ANNs

a limited training database, the hybrid discrete HMM/MLP approach always outperforms the standard discrete or multi-Gaussian HMMs.

Lippmann and Singer (Lippmann & Singer, 1993) investigated two approaches to integrating neural network and HMM for word spotting applications. In one approach NN were used to perform second-stage postprocessing on the putative hits produced by a high-performance, tied mixture HMM word spotter. A second approach combined RBF, Viterbi decoders and word-level back-propagation training into an integrated primary word spotter.

Bengio (Bengio *et al.*, 1992) also used MLP's as the probability estimator. The ANN is first trained to approximate recognition of phonetic features while an HMM is trained with the Baum-Welch algorithm using the trained ANN outputs as observations. In the last stage global tuning is performed in order to optimize parameter estimation of the whole system. The performance results can be seen in Table 4.1.

In an approach to fixed text speaker verification, a hybrid HMM-MLP was proposed by Naik and Lubensky (Naik & Lubensky, 1994) for telephone speech applications. A large database collected over the telephone network was automatically segmented using a supervised HMM-Viterbi decoding scheme. The speech data consisted of LPC cepstral coefficients, delta coefficients and the energy parameters of those mentioned features. These features were transformed into a new set of features through a discriminant technique designed to maximize the separability between the true speaker and the impostors. Then using this segmented data an MLP was trained with it. The hidden layer has 64 units while the output layer has 124 units. Through a scaling procedure, the output scores of the MLP were used as observation probabilities in a Viterbi realignment and scoring step. A significant improvement was observed (20%) based on the hybrid approach to speaker verification over the DHMM method.

4.1.1.3 Multilayer Perceptron as a Vector Quantizer

A different approach was suggested by Cerf and Campernolle. Most of the work discussed so far concentrated on the use of MLP's as the probability estimator. The main advantage of this approach (as compared to the traditional HMM) is the discriminative nature of MLP training. Cerf also points out that there are disadvantages to this approach (Cerf & Compernolle, 1993). Firstly, they are restricted to phonetic HMM and many small vocabulary applications use word models instead of phonetic models. The second disadvantage is the demand of training the network towards the global minimum. Usually this type of approach requires large networks and training can be a difficult task. In the Cerf approach, the MLP acts as a vector quantizer for the discrete HMM parameters. MLP can be trained for phonetic classification and will act as labelers for the system. The network architecture for this purpose was small. Several MLP-VQ HMM strategies were carried out and the Multi-MLP approach is the outstanding one. This approach performs as well as the discrete HMM but with fewer HMM parameters.

Finally integration of MLP with the Radial Basis Function (RBF) has been considered by (Kilmartin & Ambikairajah, 1992) has been reported. The MLP operates in the time domain and is used to extract speech parameters which are later used for verification with the RBF classifier. This MLP was trained to act as a non-linear predictor for the utterance being applied to the system. The network learns to predict the $n+1$ speech samples from the previous k samples. The back propagation algorithm was applied to the given task. The final weights in the MLP after convergence were used as an input in the next stage for classification. The second stage makes use of these inputs to perform the speaker verification task based on the RBF classifier. This classifier is tuned to accept the true speaker weight vectors and to reject the weight vectors from other speakers. Five client speakers and four impostor speakers were used in the evaluation

of the system. The hybrid approach was compared with the MLP SV carrying out the same task. It was found that the MLP SV approach requires more hidden units and longer training time in order to achieve the same level of performance.

4.1.1.4 Multilayer Perceptron Implementation of HMM

In the connectionist approach of HMM applied to speaker verification, two models of HMM are trained where one model is speaker specific and the other model is trained with a large population of speakers. This pair of models is treated as a connectionist network termed as alpha-nets (Carey *et al.*, 1991). This approach utilized the discriminatory abilities of MLP with HMM. This allows the HMM to have a discriminative training capability. The errors are propagated back from the outputs of HMMs and used to modify the parameters of the model to increase discrimination. This approach obviates the need to set an absolute threshold since acceptance of the utterance is based on the difference between the matches for the competing models. Discriminative training has also been shown to increase the performance of the system. In a real time implementation of the system, an average digit error rate of 4.5% was obtained with only one mis-classification in 600 trials using 5 digit sequence.

Neural networks can be an efficient solution for speaker recognition because they have powerful discrimination abilities. However, neural networks cannot deal fully with the problem of the time variability of speech. In the next chapter, a method is proposed to enable the neural network to improve acceptance of time alignment inputs. On the other hand, time alignment is efficiently handled by traditional methods such as DTW and HMM. The above reviews have seen several approaches trying to combine DP or HMM with NNs in application to speech or speaker recognition. The area of research for hybrid approaches can be classified as either connectionist implementation of HMM or cooperation of two methods, taking the advantage of

the characteristics of each method as a way to enhance the performance.

4.1.2 HMM-MLP Speaker Verification System

This section presents details of speaker verification experiments using a new method based on Multi Layer Perceptrons (MLP) combined with hidden Markov models (HMM) for telephone speech. The output scores of the HMM were used as the inputs to the MLP. The method achieves improved performance by the use of more than one feature set for each set of pre-processed parameters. This uses the fact that different feature sets can produce different partitionings of the vector space. The speech parameters used are the cepstra and delta cepstra. It was found that further improvement of the verification method was achieved with the addition of the MLP. The results show that the HMM-MLP combination achieved significant equal error rate reductions when compared to the traditional HMM.

Previous work (Forsyth *et al.*, 1993) used the standard HMM with a fixed transition probability to model temporal features. A discrete hidden Markov model (DHMM) was constructed and re-estimated to model the spectral features of telephone speech data. Since duration information can assist in discriminating between speakers, so state duration modelling using Gaussian duration probability density functions was used in DHMM. Figure 4.1 shows the proposed speaker verification method of the combined HMM and MLP. The cepstra and the delta cepstra features were used to train the HMM models resulting in two output probabilities. Two feature sets were used to improve the performance of the verification method. The first stage of evaluating the performance of the verification method is to combine the scores from the two models which is termed as the "old decision". The aim of this work is also to evaluate the suitability of incorporating neural networks for the verification task. The log likelihood outputs from the HMM models are fed to the neural network which is trained with the back propagation algorithm. Multiple

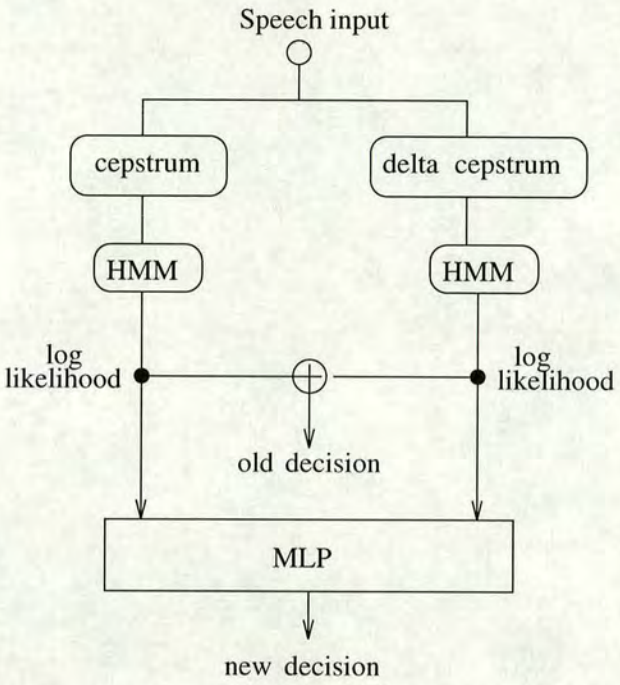


Figure 4.1: Combined Speaker Verification System- an MLP Classifier Applied to Aid the Final Decision Making.

feature sets with the two information streams now have a new verification score which is termed a "new decision". The result obtained was then compared with that of the "old decision". A database of twelve isolated digits (digits 'one' to 'nine' plus 'zero','nought' and 'oh') was used in the experiments.

HMM verification experiments were performed by first creating digit models from the speech data. Before the recognition phase was carried out, a codebook of size 32 was created to represent the digits spoken. The probability of the observation sequence from the HMM was computed using the Viterbi algorithm. Two series of experiments were carried out with the HMM system using two commonly used feature sets, LPC cepstra and delta cepstra for the speaker modeling stage of the verification process. These speaker trained models were evaluated with the test data to estimate false acceptance (FA) and false rejection (FR) rates from the same speaker and also from all other speakers. The FA is when the impostor is mistaken as the client while FR

is the rejection of a genuine speaker. The equal error rate (EER) is when the proportion of FA equals that of FR. Results are presented which show how such measures can be used to give a more meaningful representation of a system. These measures (as well as some others) will be discussed in more detail in the following chapters. The use of a single feature set for cepstra and delta cepstra was taken from (Forsyth *et al.*, 1993). By using a simple linear sum, it is possible to combine the different information streams and obtain a decision (old decision) for the verification method. By combining cepstra with delta cepstra a speaker independent EER of 0.81% has been achieved. The combination of the information streams is a task for which the neural network is best suited. A further experiment was carried out to determine if the neural network can further improve the performance of the verification score.

There were 11 client speakers and 93 impostor speakers. The database was divided into 5 blocks each containing 5 tokens per word. If the first block of 5 tokens is used for training then the remaining 20 tokens are used for testing. The verification score was extended over multiple digits. The scores from several digits from the same speaker were computed for digit sequences. If the length of the digit sequence is L then the sequence consists of the first L digits from the given database digits 'one' to 'nine' plus 'zero','nought' and 'oh'. The 12 digit sequence score was obtained from each client speaker. Results were collected over all speakers to obtain 2,200 true client scores and 4,960 impostor scores from the different parameter sets. Some data sets (495 true scores and 495 impostor scores) were used to train the network and the remaining unseen patterns were used as the validation process. In this experiment the learning rate was set to 0.1 and the momentum to 0.9, the weights were corrected after the presentations of all the training data. The neural network was trained until the total error E at the output units fell below a pre-defined value. The number of input, hidden and the output units were 2, 5 and 1 respectively. From observations, the use of less than 5 hidden units gives poor performance while there was no significant improvements obtained by using more than 5 units. A combination of HMM

HMM-MLP	HMM		
cepstrum + delta cepstrum	cepstrum	delta cepstrum	cepstrum + delta cepstrum
0.56%	1.51%	1.88%	0.81%

Table 4.2: Comparison of EER for Speaker Verification Options.

and MLP networks shows further improvement in speaker discriminative ability for telephone speech. This approach achieves 0.56% EER performance. The use of a single feature set for cepstra and delta cepstra provide an EER of 1.51% and 1.88% respectively (Forsyth *et al.*, 1993). The scores of the verification systems are shown in Table 4.2. As can be seen in Table 4.2, the worst performance came from the DHMM model trained with delta cepstra features with an EER of 1.88% and the combination of the feature sets clearly shows better performance results at 0.81%. Figure 4.2 shows a typical plot of Type I and Type II error curves obtained from the hybrid approach. Different threshold values of EER can be obtained for different experiments. Thus, the hybrid approach provided a more optimum solution for the combination of the information streams.

Another performance measure that can be used to evaluate the speaker verification system

HMM-MLP	HMM		
cepstrum + delta cepstrum	cepstrum	delta cepstrum	cepstrum + delta cepstrum
10.5%	24.5%	45.1%	11.1%

Table 4.3: Comparison of the Conventional HMM and the Conventional Model with the MLP Added. 12 Digit Sequence Length ZFA for each System.

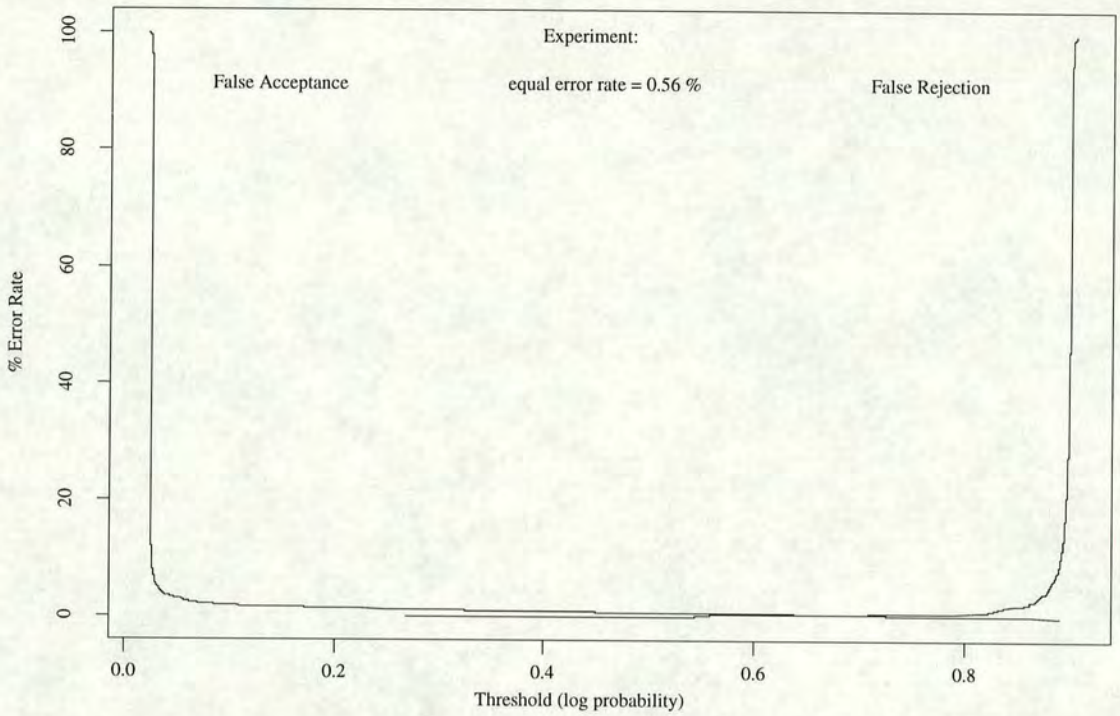


Figure 4.2: Type I and Type II Errors Achieved by the Hybrid Approach.

is the zero false acceptance rate (ZFA). In an application where high security is required this performance measure is crucial to the acceptibility of the system. The ZFA is the false rejection rate with a threshold being set so that there are no false acceptance errors. Table 4.3 shows the comparative ZFA rate performance of the conventional model against the ZFA for the same database when MLP is added to the speaker verification system. These results show that MLP is a useful addition to an HMM system.

This section has characterized the classification power of a hybrid approach to discriminate between valid clients and impostors in an ASV scheme. The hybrid approach gave a further reduction of 25% EER compared to the classical HMM. The combination of HMM and neural network in this way takes advantage of the powerful learning algorithm provided by HMM as well as the high classification power of patterns by neural network. The use of two information streams with the hybrid approach successfully improves the verification score. These results

also support the possibility of using more than two information streams for further research.

The aim here was to compare the neural network approach, and more precisely the multi layer perceptron (MLP), to the hybrid approach of SV. The above method is compared to the NNM-C from the viewpoint of robustness against relatively small amounts of training data and use of an efficient preprocessor to the neural network. This model requires the design of a client codebook and uses the codevector as well as the distortion value as the inputs to the MLP. In NNM-C, the C stands for client codebook. The next chapter provides the details for the design of the NNM-C SV system. There is 20% reduction in error rate with the HMM-MLP method when compared with NNM-C. The HMM-MLP method is more robust than the NNM-C method given the small amount of training data and a preprocessing stage with multiple features proves useful in the design of the verification system. For the HMM-MLP method, each model is trained independently. In this model, the HMM will try to optimize the production probability of the time sequence and this will improve the discriminative ability of a NN classifier at the next stage since the HMM preprocessor prepares each class to be distinguished from each other. The above method is also not free from drawbacks, since inspite of improved performance, the complexity of the SV system increases when compared with the NNM-C. It is much easier to develop a vector quantizer preprocessing stage because of its simplicity compared to HMM as the preprocessor for the SV system. Care must also be taken when multiple information streams are added to the system. The computation involved can increase significantly at the feature extraction stage as more features are added into the system. For example, the addition of delta cepstra feature requires minimum computation when compared with SV system that used LPC cepstra and MFCC together.

Chapter 5

NEURAL NETWORK MODELS FOR SPEAKER VERIFICATION

5.1 Introduction

Vector Quantization (VQ) has been successfully applied to speaker recognition. The system is based on constructing highly efficient short term spectral representations of the client speaker using vector quantization codebook construction techniques. Some techniques are based on long training sequences of speech, dividing the speech into frames and using a clustering algorithm to generate the codebook. Others take advantage of the parallel processing of neural networks.

One of the most popular codebook training methods is the (Linde, Buzo and Gray) LBG algorithm. There are two techniques used to develop the codebook. The first technique starts with a codebook of the correct size. This is known as the k-means clustering algorithm (Linde *et al.*, 1980). The algorithm begins by selecting arbitrary L vectors as centroids (L being the codebook size). The training data is distributed to distinct regions using a distortion measure. The second technique starts with a simple codebook and builds to a large one. The building of the cells makes use of binary codevector splitting. In this, the centroid for the whole set of training vectors is first found and this single codevector is then split into two codevectors. So, the initial codebook starts with dividing the training vectors into these codevectors and then finding the new centroids of the cells. The new centroid would be the new codevector. The training sequence is once again computed to find the distance between each vector in the

training sequence and the new codevectors. The codebook distortion is computed by summing the distances between the vectors allocated to each cell and the cell centroid across all the cells. If the codebook distortion value is better than the previous distortion value, then the complete training sequence is again used to further adjust the codebook. The procedure is repeated until the reduction in the overall average distortion between consecutive iterations falls to some small threshold value.

Neural networks provide another alternative to codebook design (Wu *et al.*, 1991) (Oscal *et al.*, 1992). The neural network approach is ideal for speech processing due to their massively parallel connection structures and self organization learning schemes. With the advantages of parallelism, neural networks can be implemented with the use of parallel processors which offer the potential for real time VQ. There is also a large body of training algorithms that can be adapted to yield a better VQ codebook design. In an application where the source statistics are changing over time then neural networks can be adaptable. In the work carried out in this thesis, a self organization network is combined with the LBG technique to design the vector quantizer. Once the codebook is generated, the preprocessing stage measures local similarities by using a vector quantizer to select the index. The indices of the winner nodes are fed to a neural network classifier in which the system can be trained and evaluated. In real applications of speaker recognition the amount of data available is strictly limited by what the client can offer. The amount of training data required is task dependent. The critical application is for telephone speech where the design objective is to obtain the highest possible performance from a very limited amount of training data.

For any given database there will be a variation in performance among the speakers. Some speakers have voices that are distinctive and have no difficulty with the SV systems. The false acceptance rate in this case will be low. Other speakers have difficulty using the SV systems as they might have common voice characteristics. In theory neural networks should be able to produce the desired outputs from any input representation that encodes the relevant information. In practical cases, an optimal input representation and preprocessing is normally required for

an efficient network. For each client speaker the codebooks are produced from that speaker's own training tokens. The codebook design in this case may well represent the distribution of speech features for each speaker but will not discriminate the different characteristics among different speakers. The work addressed here will examine and compare the use of NN SV that uses separate client codebooks for different digits and separate impostor codebooks for different digits. It is assumed that including these impostor codebooks in the training data of each network will increase the efficiency of this data to direct model the differences between the client speaker and impostors. For example, client tokens that have similar values using the impostor codebooks are highlighted and vice versa.

It is important to note that data preparation can make the difference between a network that performs well and a network that performs excellently. If the improvement in performance is significant then this addition of impostor codebooks would be acceptable. The preprocessing stage that makes use of codebooks from the client and the impostors is the subject of further experimentation. In practice, it is often necessary to experiment with a variety of input representations.

The technique of hidden Markov models (HMM's) is one of the popular methods for speech recognition and has been successfully applied to speaker recognition (Che & Lin, 1995). In conventional approaches such as VQ and hidden Markov models (HMM) (Buck *et al.*, 1985) there is a lack of explicit discrimination between classes. These approaches make use of data from only the client speakers. In recent years there has been growing interest in using neural networks for speaker recognition. Neural networks have been used in many applications and have proven to be able to generalize on unknown samples. One type of network that is commonly used is the multi layer perceptron (MLP). Research for speaker recognition using neural networks is still a relatively young field. In one study, Hattori (Hattori, 1994) used a predictive neural network. It non-linearly predicts the next frame from several preceding frames. This predictive neural network is based on the MLP (Multi Layer Perceptron). In a text-dependent SV an equal error rate (EER) of 1.5% was achieved for 12 male speakers. (Jou *et al.*, 1990) proposed a text dependent speaker verification (SV) using one MLP per speaker. Oglesby and

Mason have proposed using MLP and Radial Basis Function (RBF). The performance of the system is greatly affected by the training tokens as well as the network architecture. One of the main problems faced using these methods is that it requires large number of hidden units and the training can be time consuming (Oglesby & Mason, 1990)(Oglesby & Mason, 1991). These approaches used raw features to feed into the neural network. The larger and more complex the input space the more training samples are needed for training before the network can learn to generalize (Waibel *et al.*, 1989)(Maren *et al.*, 1990). There is also the possibility that large number of hidden nodes are required to solve the problem. If this is the case then training may be difficult as not only will the MLP take a long time to train but it will also increase the possibility of being trapped in a local minimum which may not yield a good solution to the problem. In this thesis combined neural network approaches makes use of a preprocessing stage which has three advantages. Firstly, since each word has its own codebook, this simplifies the vocabulary reconstruction. New words can be simply added to the system. The number of codevectors can be modified at will while other codebooks remain unchanged. Secondly, each small network can be trained separately, this gives the possibility to expand the classifier without an increase in computational effort. Finally, the preprocessing stage allows a smaller network configuration. This can eliminate the difficulties in the training phase and facilitates training on limited data.

This chapter is organised as follows. Section 5.2.1 describes the learning algorithm and the classifier for the SV system. Section 5.3 describes the database. The number of layers, input and hidden units are discussed in section 5.4. The choice of parameter values, such as the momentum term and learning rate used by the MLP were also discussed and experimental results are quoted to show how the choice of these parameter values influences the performance of the MLP. Also, included in this section are the experimental results from NNM-CI and NNM-C SV systems. In section 5.5 the temporal structure of speech is addressed for the static classification approach. Finally, section 5.6 provides performance data using speaker independent thresholds.

5.2 The Learning Algorithm

This section describes the unsupervised and supervised learning used in speaker recognition. The unsupervised learning considered here is self-organized learning. The supervised classifier is the MLP network.

5.2.1 Self-Organization Learning

Speech patterns in N-dimensional space which are similar in some respect to one another can be clustered together on the basis of class membership. For example, speech patterns belonging to class C_i might be clustered closer to one another in preference to any pattern belonging to class C_j. Unsupervised learning techniques such as the self-organization network try to identify the speech patterns as cluster centres. This learning scheme can be used for efficient representation of speech feature vectors. A self-organized network consists of the input layer and the output competitive layer. In this network, the codewords W_i associated with the units are initialized to small random values. The training algorithm iterates by selecting the winning unit N_i and adjusting the weight W_i after each presentation of the input vector **X**. This algorithm was studied by Kohonen (Haykin, 1994) and variations of this algorithm have been used by (Wu *et al.*, 1991)(Calonge *et al.*, 1995) in speech recognition problems. The algorithm for the codebook design is executed in two stages: the initialization and optimization.

Codebook Design

Step1: initialization

Set j=1 and initialize all the weight vectors (initial codewords) W_i (i=1,2...L) with the first L input vectors where $W_i = X_i$ and $W_i = [W_{i1}, W_{i2}.....W_{ik}]$.

Step2 : classification

For each input vector X_p calculate the Euclidean distance of the nearest codeword to each data

vector.

$$d_{ip} = \sum_{n=1}^{k} (X_{pn} - W_{in})^2 \tag{5.1}$$

Select the output unit N_{i*} with the smallest distortion and label it as the winner.

Step3 : update the winning weight W_{i*} with X_p

$$W_{i*}(new) = W_{i*}(old) + \delta[X_p - W_{i*}(old)] \tag{5.2}$$

where δ is a gain term and decreases as training progresses.

Repeat step 2 and step 3 for all training vectors.

Step4 : distortion measures

Calculate the average distortion D_j as

$$D_j = \frac{1}{Q} \sum_{n=1}^{Q} (\min d_{in}) \tag{5.3}$$

where Q is the number of input vectors to be quantized.

Step5 : if $(D_{j-1} - D_j)/D_j < \epsilon$, go to step 6;

otherwise repeat step 2 to step 5 with $j = j + 1$.

Here ϵ is a small distortion threshold.

Step 6: the complete set of L codewords called the codebook is obtained.

5.2.2 MLP as Classifier

Basically the neural network considered here is a multi-layer perceptron. A three layer back propagation network will have an input layer, an output layer and one hidden layer. Experiments conducted by Moore, Peeling and Varga showed that poor performance was achieved using

two hidden layers as compared with one hidden layer (Peeling *et al.*, 1987). Since there is no theoretical basis for the number of hidden layers to be used, this method uses one hidden layer. The feedforward network has all connections and data flows from input layer to output layer. The neural network was trained using supervised learning. In other words, for each input pattern there is a known desired output. During training, data is compared with the desired output to derive an error that is propagated backwards for the weights being changed. The activation function used here is a non-linear function called the sigmoid function which determines the values on output nodes. An MLP can have several outputs, each of which represents one of the given categories. In this case the MLP acts as a discriminator which decides whether an input pattern belongs to the given class or not. The goal of the training process is to minimise the error over all the training patterns. The error for a given pattern is the summation of the difference between the target and the output over all the output nodes for that pattern. The error is propagated backward for proper adjustment of the weights in each layer.

For an output unit the weight changes:

$$W_{jh}(new) = W_{jh}(old) + \eta\delta_j O_h + \alpha[\Delta W_{jh}(old)] \tag{5.4}$$

For a hidden unit the weight changes:

$$W_{hi}(new) = W_{hi}(old) + \eta\delta_h O_i + \alpha[\Delta W_{hi}(old)] \tag{5.5}$$

where

O_i - the output of the input layer

O_h - the output of the hidden layer

δ_j - error signal at the output nodes

δ_h - error signal at the hidden nodes

η - learning rate

α - momentum term

$\Delta W_{jh}(old)$ - previous weight change between unit j and unit h

$\Delta W_{hi}(old)$ - previous weight change between unit h and unit i

Before the weights are updated, the MLP starts with randomized values. The process of testing then is to present patterns not used during training. The network is said to generalize, if good performance is achieved from the verification test data. A detailed description of the back propagation algorithm can be found in (Rumelhart & McClelland, 1986).

Various approaches have been developed for whole word based SV tasks. In these cases word variations may occur especially in repeating the right intonation. One form of variability that can affect the word based recognition is the non linear compression and expansion of the speech signal from one word to the other. In an application for speech recognition or speaker recognition it is necessary to time align the input vectors which contain the spectral information before being fed to neural networks. For example, in one approach a hybrid neural network which consisted of a Kohonen map and MLP was proposed for speaker independent isolated word recognition. The 12 LPC derived cepstrum parameters were extracted from each speech frame which was used as the parameter vector. Each frame was 37.5 milliseconds long, and with a linear time warping, the overlaps between the adjacent frames were set so that the total number of frames is 41 regardless of the word duration. The final outputs of the Kohonen map which were used as the feature vector were fed to the final stage of the recognition system for classification. The classifier receive eighty two indices obtained from the Kohonen map as features to its input layer. These features belong to the 41 consecutiveye frames. The system achieved 93.82% accuracy on 11 Farsi numbers (Tabatabaee *et al.*, 1994).

A more efficient procedure relies on the application of dynamic programming. Dynamic Programming Neural Networks (DNN) have been proposed for speech recognition tasks using dynamic programming (DP) and multi-layer perceptrons. The input units are arranged in a block structure frame along the time axis. The input pattern is optimally time aligned by DP so that the output unit gives maximum output. A speaker independent isolated Japanese digit recognition experiment was carried out with 107 speakers resulting in 99.3% recognition accuracy. This

DNN made use of the valuable features of time normalization of DP and classification power of NNs (Sakoe *et al.*, 1989).

Another procedure to normalize the length of the word is the use of trace segmentation. This procedure was used by Zhu and Fellbaum (Zhu & Fellbaum, 1990) to compress the non-linearities of the speech signals into fixed lengths of 24 triples. The outputs of the MLP give the classification results. The system was capable of achieving 95% accuracy from 22 German words used in the experiments.

From the above examples, the simplest approach to overcome the variability of the speech is the linear time normalization (LTN). LTN is made to correspond as closely as possible to a straight line joining the initial and final points. The approach of using LTN is implemented in the work carried out in this thesis for the verification system. The technique presented is a simple procedure of omitting or repeating the frames to normalize a variable length set of features with a fixed number of parameters.

The verification system is shown in Figure 5.1 and Figure 5.2. The initial stage used the commonly used feature set, cepstral coefficients for the speaker modelling stage as the speech signals. The SV system is made up of two phases : the preprocessing vector quantization (self-organization network) and the MLP classifier. The codebook design was described in the previous section. Each codebook contains 32 codewords and there are 12 of such codebooks [1] for each client speaker in the system. With an efficient preprocessing vector quantization, the indices of the winner nodes are fed to the second stage of the verification system for classification. The classifier system is based on a three layer perceptron trained using the back-propagation algorithm (Lippmann, 1987). This back-propagation algorithm is probably one of the best approaches to use if the input array is a few hundred units or less (Maren *et al.*, 1990). The type of function used at the output node is the sigmoid function.

[1] one for each digit including "zero", "nought" and "oh"

In NNM-C, the output of the preprocessor for each time frame would contain the index j of the codevector with the minimal distortion and the corresponding distortion value d. In NNM-CI, there are two pairs (j,d) per frame, one for the client codebook and one for the impostor codebook. The input pattern is linear time normalized (LTN) either by linear compression or expansion so that the total number of frames becomes a constant regardless of the word duration. Through this preprocessing, the highly redundant speech data are reduced so that only the useful information regarding the codevector and the distance measure is retained in the feature vector to feed the MLP. For example, if the number of frames after LTN equals 40, two coefficients per frame will fit the 80 input units. By using the client codebook and the impostor codebook each hidden unit is fed with 160 input units resulting in an architecture of 160-N-1 where N is the number of hidden units. The output unit is trained to respond with a high probability value (0.99) output for the desired speaker and a low value (0.01) for the other speaker data. The training scheme used a separate network for each digit for each speaker. Separate networks were trained for each of the 12 digits for each of the 11 speakers.

5.3 Speech Data

The database consists of isolated digits from a large number of speakers. Twelve isolated digits (digits 'one' to 'nine' plus 'zero','nought' and 'oh') were used in the experiments. A group of 11 speakers are modelled by the system and an independent set of 83 impostor speakers is used for testing. The data are all end-point detected to remove excess silence and minimize storage requirements. The frame size was 20ms with 15ms overlap. The training templates consisted of 5 tokens from the client speaker and 19 tokens from impostors who were different from the impostors used in testing. The templates from the target group and the impostor group were alternated in the training set. The implemented verification system used another set of data (not used during training) for further evaluation of its performance. It was tested on 20 true speaker tokens and 83 impostor tokens for each digit for each speaker.

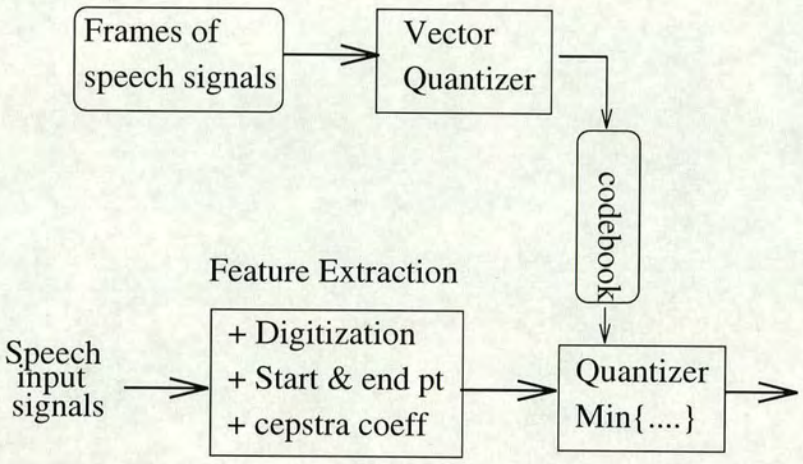


Figure 5.1: Preprocessing Unit Block Diagram for Unsupervised Neural Networks

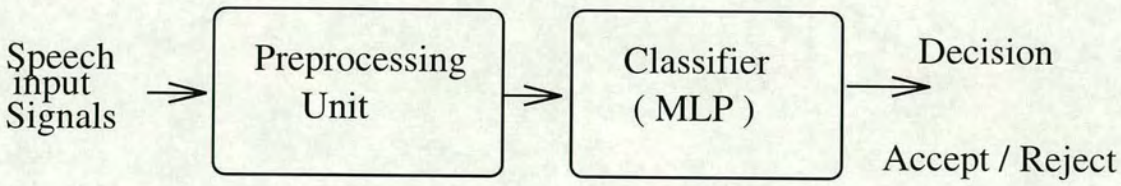


Figure 5.2: Neural Network Based Speaker Verification System

5.4 Experimental Results

The task of speaker verification is to verify true client from impostor speakers. If the network is trying to determine the presence of a client, the sum square error has little intuitive meaning in evaluating the performance of the network. The global success rate of the verification system is thus evaluated as the mean value of the success rates obtained with each speaker. A test token can be accepted or rejected if the output score is respectively greater or lower than the predetermined threshold. Thus, two kind of errors are evident: falsely reject (FR), when the speaker is a valid client but wrongly rejected and falsely accept (FA), when an impostor is mistaken as a client. In the evaluation of the verification system the use of equal error rate (EER) thresholds means that all thresholds are determined a *posteriori*. This approach sets the proportion of FA equal to the proportion of FR resulting in the said EER. The zero false acceptance (ZFA) rate is the false rejection rate with a threshold being set so that there are no false acceptance errors. The zero false rejection (ZFR) is the false acceptance rate where no genuine speakers are rejected. In an application where a negligible false rejection rate is important to the acceptability of the SV then ZFR rate is a more useful measure than the EER. On the other hand if the security is high then the ZFA performance measure is more appropriate to the system. These performance measures will be discussed in more detail in chapter 7 for evaluation of the different SV techniques. This section describes the experimental results related to aspects of the variation of network parameters, network complexity and the performance of the verification system.

5.4.1 Variation of Network Parameters

Generalization of a given network is influenced by the size and the efficiency of the training data as well as the architecture of the network. In the problem presented, the set of training data is fixed and the issue of interest is that of determining the best architecture of network for generalization. Researchers working on speaker dependent speech recognition systems have reported good results using a large momentum term and small learning rates. Some of these parameters require more presentation during the training phase before the network generalised. (Rumelhart & McClelland, 1986) specify the values of 0.1 and 0.9 as standard for the learning

rate and the momentum term respectively. However, not all practitioners follow this advice as certain parameters might perform well for a given network in a particular task and might be worse for other network structures.

An initial experiment was conducted to investigate the effect of different values of learning rate and momentum term. The training data was fed into the network cyclically and the number of hidden units was fixed to 10. Network weights were initialized with three different random values uniformly distributed from $\pm 0.05, \pm 0.1, \pm 0.3$. The experiments were evaluated with the test sets from 5 client and 83 impostor speakers. It was tested on 20 tokens from the client speaker and 83 impostors' tokens for each digit for each speaker.

Using the back propagation algorithm the network was trained with a fixed learning rate (0.05, 0.25, 0.45) and a fixed momentum term (0.45, 0.65, 0.95). A summary of the results obtained is shown in Table 5.1. It is evident from the results that large values of learning rate and momentum term give the worst results. It is also found that optimal parameters have to be individually determined to give optimal results where the result of each pair of learning rate and momentum terms gives the smallest number of errors. Variation of the learning rate and momentum term does noticeably influence the performance. The best result for these experiments is derived from using a small learning rate of 0.05 or 0.25 with a momentum term of 0.95.

		DIGITS EER (%)												
LR	MT	1	2	3	4	5	6	7	8	9	10	11	12	mean
0.05	0.45	26.5	33.0	24.1	20.7	23.7	26.1	19.2	28.3	25.2	24.5	27.3	22.1	25.0
	0.65	27.4	32.8	23.6	20.7	23.6	25.5	19.2	27.6	26.0	25.7	27.3	21.1	25.0
	0.95	24.0	26.5	22.8	19.2	23.4	23.8	18.3	24.1	22.3	23.5	25.1	28.5	23.4
0.25	0.45	27.4	34.3	24.6	20.2	23.0	25.6	19.5	27.3	24.3	23.5	27.6	22.7	25.0
	0.65	27.0	34.1	25.3	20.2	22.6	26.2	20.3	27.8	24.3	24.3	27.4	22.4	25.1
	0.95	25.0	29.9	22.0	20.5	23.6	25.7	20.7	25.9	23.4	23.2	25.5	22.0	23.9
0.45	0.45	27.0	34.5	25.0	20.5	23.2	25.9	20.1	28.5	25.2	23.5	27.4	22.9	25.3
	0.65	28.5	34.3	24.6	20.6	23.1	25.1	20.1	29.1	26.1	22.9	27.2	22.4	25.3
	0.95	34.4	37.0	29.2	27.7	24.1	27.9	23.0	31.1	29.2	28.3	30.2	23.3	28.7
LR - Learning Rate							MT - Momentum Term							

Table 5.1: Average EER from 5 Client Speakers for MLP with 10 Hidden Units.

		DIGITS EER (%)												
		1	2	3	4	5	6	7	8	9	10	11	12	mean
Hidden Units	2	26.4	25.9	19.0	7.3	15.3	13.0	12.3	29.4	22.2	15.3	19.1	5.5	17.5
	4	26.5	20.9	17.1	6.8	12.2	14.8	12.9	30.6	22.0	15.3	15.4	4.9	16.6
	6	31.3	17.7	18.3	11.0	12.2	11.8	11.1	26.9	20.9	11.0	16.6	6.1	16.8
	8	30.1	22.7	20.2	9.3	11.6	14.1	10.5	27.5	20.3	11.6	17.8	2.4	16.5
	10	29.4	19.6	18.4	8.6	14.6	13.5	11.7	23.3	20.9	17.8	18.5	2.4	16.5
	12	30.1	19.1	13.5	7.4	12.8	12.3	12.3	25.1	21.6	18.5	19.0	2.4	16.1
	15	30.1	20.2	16.5	9.3	12.2	11.1	11.1	27.6	22.1	19.0	17.8	3.0	16.6
	20	30.4	21.4	14.1	8.7	12.9	9.9	12.9	25.7	23.3	15.4	17.8	5.5	17.8
	25	30.7	27.5	19.6	8.7	15.9	14.8	12.7	25.2	22.3	15.4	19.1	3.0	18.8

Table 5.2: EER for Client Speaker (E) with Varying Number of Hidden Units

5.4.2 Network Capacity

The capability of the network to capture the underlying characteristics of the client data relies heavily on the number of hidden nodes. If the hidden nodes are insufficient there is limited freedom to form the decision surface. More hidden nodes would then provide the extra flexibility for the network to converge. However, care must be taken not to exceed the limit rendering generalization impossible. Since there is no theoretical guideline for this problem, several experiments were conducted to determine the number of hidden nodes for this task. The effect of varying the number of hidden nodes is considered for a single layer network. The number of hidden nodes used in the experiments are varied from 2 to 25.

In order to assess the effect of hidden units of the MLP, the learning rate was set to 0.05 and the momentum term as 0.95. Beginning with two hidden units, the number was increased by two until there were 12 hidden units. After this point, the number of hidden units was increased by 5 each time up to a maximum of 25 units. Table 5.2 shows the effect of varying the hidden units over the test set. All experiments show a decrease of EER with an increase of hidden units to a certain point. No further improvement can be seen as the number of hidden units increased after this point. This can be attributed to the limited size of training data used to train the MLP. With only 5 training tokens, the excess hidden units will prevent the network from to generalising. Rather than learning only the general patterns to produce the correct decision, the network learns (to excess) the individual samples. If more training tokens were used, improved performance of the EER would be expected over the wide range of the hidden units before settling to a fixed point. High performance across various digits will differ among speakers. For example, with four hidden units digit 4 from client (E) provides an EER of 6.8% whereas for another client speaker (F) the EER is 17.1%. As long as the number of hidden nodes is appropriately selected then generalization can be expected when tested with the unseen data. The number of hidden units selected for the different SV systems are given in section 7.4.2 of Chapter 7.

		EER (%)			EER(%)
DIGITS	one	19.7	Number of Digits in Sequence	1	19.7
	two	23.8		2	20.2
	three	19.1		3	14.4
	four	14.5		4	10.1
	five	14.1		5	7.9
	six	20.0		6	7.5
	seven	15.8		7	6.6
	eight	20.0		8	5.9
	nine	15.2		9	4.5
	zero	16.9		10	3.9
	nought	22.6		11	3.9
	oh	13.0		12	3.0
	mean	17.8			

Table 5.3: Single Digit Results Summary and Digit Sequence Results Summary (NNM-CI, LTN = 40)

5.4.3 System Performance

As mentioned earlier the evaluation of the verification system used the EER. In addition to the EER results based on single digits, average results are also quoted for error rates that combined the results for all of the twelve digits. Combining the scores from several digits can further improve the SV performance. The combination of these scores is commonly referred to as using a digit sequence.

5.4.3.1 Digit Sequence Results with Fixed LTN (NNM-CI)

Here 11 speakers were chosen as a set of client speakers and were enrolled in the SV system. The SV system is evaluated on single isolated digits and digit sequences with fixed LTN. The length of the input is 40 after LTN. The EER was evaluated for each of the 12 digits. The performance of each digit is listed in Table 5.3 with the EER ranging from 13% to 24%. This is the average EER for each of the digits. It is obvious from the results that there is considerable variation in performance among the various digits. This variation may be due to differences in the amounts

of speaker discriminating information embedded in various digits.

The verification score is extended over the various digits using digit sequences. In order to make the verification decision, a common threshold is applied to the verification score taken over all the digits. Eventually, improvements in the speaker discriminating information of the digits becomes apparent as more digits are added. The performance of the digit sequence results is shown in Table 5.3. For a network with 40LTN from the 11 speakers the average EERs are 17.8% on a single isolated digit and 3.0% on a sequence of 12 isolated digits.

5.4.3.2 Digit Sequence Results With Fixed LTN (NNM-C)

Table 5.4 shows the average EER using the NNM-C. The average EER for the single digits is 13% with a range from 7% to 16%. The EERs compared graphically for the 12 digit sequences shown in Figure 5.3 were 1.04% (NNM-C) and 3% (NNM-CI). The NNM-C performs better than the NNM-CI for all of the digits. The performance of the NN models is linked with the hidden layer of units with its ability to generalize. The knowledge which is stored in the hidden layer is abstracted from the information contained in the input patterns. Each hidden unit will respond to the different input patterns presented to it. The NNM-CI design works but is not optimal for two reasons. First, in this model the decision surface may be more complex due to the complexity of the input to the classifier. On the other hand, NNM-C has a simpler feature vector representation which increases the likelihood of convergence to a working set of connection weights. Second, the input layer has the largest number of connections and the first hidden layer is often the largest computing layer in the network. This may require more training data before settling to a set of weights that would provide a good generalization. Both these methods were trained with equivalent amounts of training data.

DIGITS	EER(%)	Number of Digits in Sequence	EER(%)
one	13.5	1	13.5
two	14.3	2	9.6
three	13.1	3	6.9
four	15.0	4	5.0
five	10.0	5	4.1
six	15.3	6	3.0
seven	14.4	7	2.6
eight	16.2	8	2.4
nine	6.9	9	1.8
zero	8.9	10	1.3
nought	15.6	11	1.1
oh	12.9	12	1.04
mean	13.0		

Table 5.4: Single Digit Results Summary and Digit Sequence Results Summary (NNM-C, LTN equals 40)

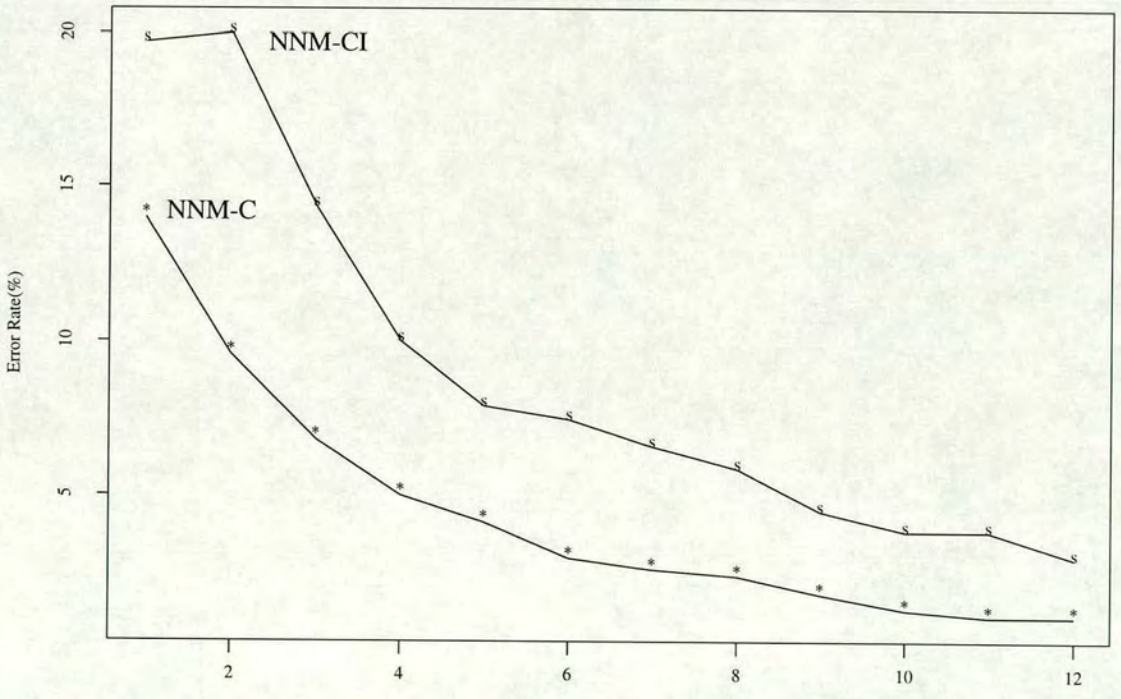


Figure 5.3: Digit Sequence EER (The Improvement in NNM-C over NNM-CI for Various Digit Sequence Lengths)

5.5 Comparing Results with Different Values of LTN

Preliminary experiments were carried out with two client speakers. To understand the temporal structure of speech in the static classification approach, different values of Linear Time Normalization (LTN) have been tested. The lengths of the inputs after LTN are 30, 40, 50 and 60. Further improvement can be seen after combining the different values of LTN. Encouraged by this result further experiments were carried out with the other speakers. This approach is simple and is a good choice to perform static verification of the speech signals with neural network.

In the previous experiments, the input patterns are time normalized by either compression or expansion to equal 40LTN. This means two coefficients per frame will have 80 input units. In this section the performance of the NNM-C model with other LTN values is evaluated. The advantages of using this model over the NNM-CI have been discussed in the previous section.

The speaking rates of the client speakers are different from one another. Each client speaker will vary their speaking rate differently and non-uniformly. In practice, equivalent words would generally have a similar pattern because of the same phonetic relationship. However, there may also be differences; words can be spoken much faster or slower, there can be differences in the vocal effort (intensity) and one word can be uttered more precisely than the other. So the same word may differ from moment to moment mainly as a result of difficulty of repeating the right intonation and timing exactly. One of the most challenging problems of the NNM-C is its fixed input size. MLP can perform a classification operation by using the fixed number of inputs without taking care of the time sequence of the speech patterns. Once the architecture is determined with the specific units and training algorithm, the network can perform the classification task for the speaker verification system. Ideally, the LTN value chosen as the inputs to the NNM-C model for this particular task is important. What is of interest here is whether increasing or decreasing the number of input units to the network would positively affect the performance. The emphasis of the experiments is on the performance of the NNM-C system under different values of LTN in an attempt to identify the best approach.

During the learning phase, the LTN word patterns from the client and the impostors are submitted to the network input layer. The dependence of the SV performance on the network training parameters (learning rate and momentum term) is to an extent based on the experience gained from the previous experiments in section 5.4.1. The experiments were carried out by using a small learning rate of 0.05 with a momentum term of 0.95. However, it should be noted that optimum parameters have to be individually determined for better performance with different network structure.

5.5.1 Single Digit Performance with Different LTN Values

In this section the performance of the NNM-C model with different LTN values is evaluated on a single isolated digit test utterance. The verification score based on the output of this model is used by applying an EER threshold to make a decision of accepting or rejecting a client. As with the previous experiments the threshold used is speaker specific. The EER was evaluated for each of the 12 digits. The performance of each of the digits used is listed in Table 5.5. Graphical presentations of the EERs are shown in Figure 5.4.

From the figures shown there are considerable variations in performance across the digits. LTN equals 30 has an average EER of 15% with a range from 11% to 18%. LTN50 has an average EER of 13.9% with a range from 8% to 18%. LTN60 has an average EER of 11.2% with a range from 7% to 15%. The best digit across the range of the utterances is zero, while digits like 1, 5 and 9 also show good performance results. The worst performances is for digits 4, 6, 7, 8 and "nought". According to Yu, Mason and Oglesby, the good performance of the digit zero can be attributed to it being the longest utterance thus containing more information as well as the fact that the voiced fricative of the first phoneme is a particularly useful phoneme in speaker recognition (Yu *et al.*, 1995). While the digits each have different performance in isolation, each digit emphasises different aspects of the time varying speech signal and the rankings of the digits may vary from client to client. The fact that specific digits can significantly improve

Digit	EER (%)			
	LTN30	LTN40	LTN50	LTN60
1	13.6	13.5	11.9	9.1
2	16.8	14.3	15.0	12.1
3	15.0	13.1	14.3	11.0
4	17.1	15.0	15.6	13.5
5	11.2	10.0	10.6	8.7
6	18.2	15.3	16.5	13.5
7	17.9	14.4	16.4	14.9
8	17.8	16.2	17.6	12.2
9	10.9	6.9	10.5	8.2
zero	10.6	8.9	7.8	6.6
nought	18.3	15.6	17.1	14.5
oh	13.5	12.9	13.1	10.7
mean	15.0	13.0	13.9	11.2

Table 5.5: Single Digit Results of Four LTN Inputs with Speaker Specific Threshold.

performance indicates that a password system consisting of these digits could be found to suit each client speaker. The performance of the digits follows the same patterns for all values of LTN used with the best overall performance from LTN60.

This section has established the relative performance of the different LTN values used in the experiments. It also suggests the possibility of selecting the best LTN values in order to improve the robustness of the NNM-C model. Detailed study of the results by each client speaker will be dealt with in the following section.

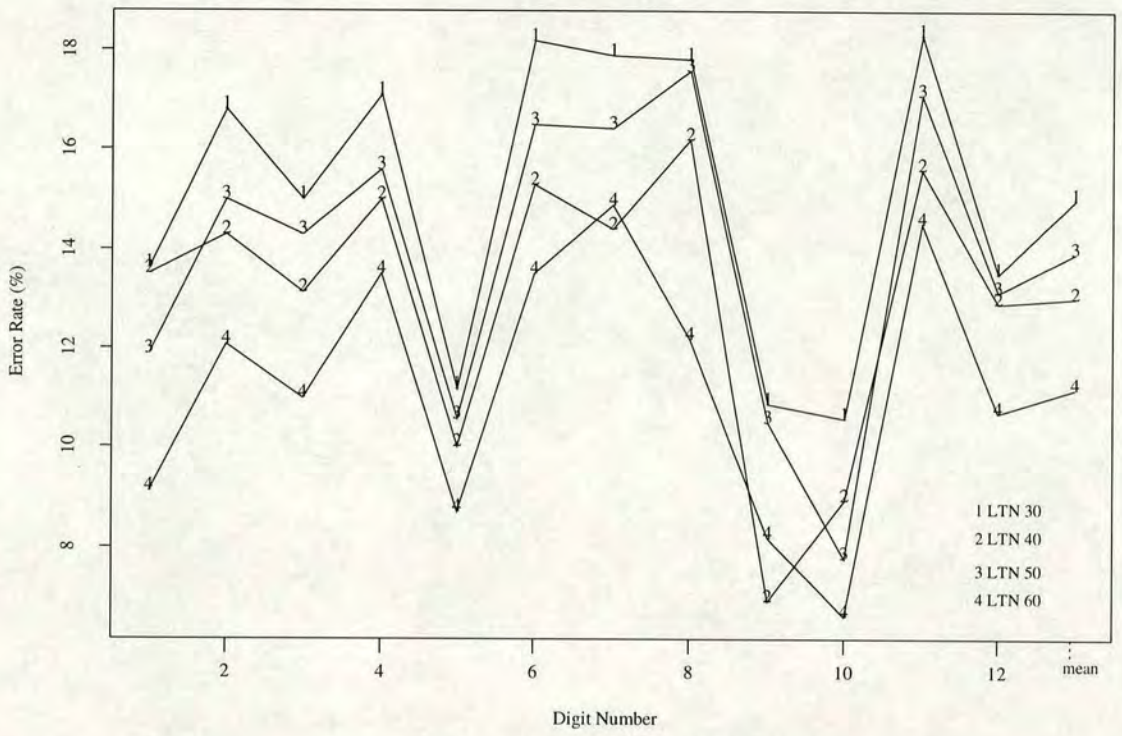


Figure 5.4: Relative Performance of Single Digits for Four different LTN. The EERs are calculated using speaker specific thresholds.

5.5.2 Digit Sequence Performance with Different LTN Values

The EER performance for each of the LTN inputs over various digit sequence lengths is shown in Figure 5.5. As more and more digits are being added, the speaker discriminative information it contains becomes more apparent. For most LTN values examined the EER appears to stabilize at around strings of eight digits. However, this is not the case for LTN30 which shows signs of instability. This is probably due to too much compression of the speech signals and some essential information may be lost which results in poor generalization of the NNM-C. An increasing number of input units does affect the verification performance but does not necessarily bring about an improvement in performance. The results suggest that proper values of time normalization of the speech signals are also needed in addition to the features used to improve performance of the speaker verification system. Ideally, knowledge stored in the hidden layer of the NNM-C model should be abstracted from the information contained in the LTN input speech patterns. This abstracted knowledge will provide the basis for the model to classify the pattern into an accept or a reject category at the output unit.

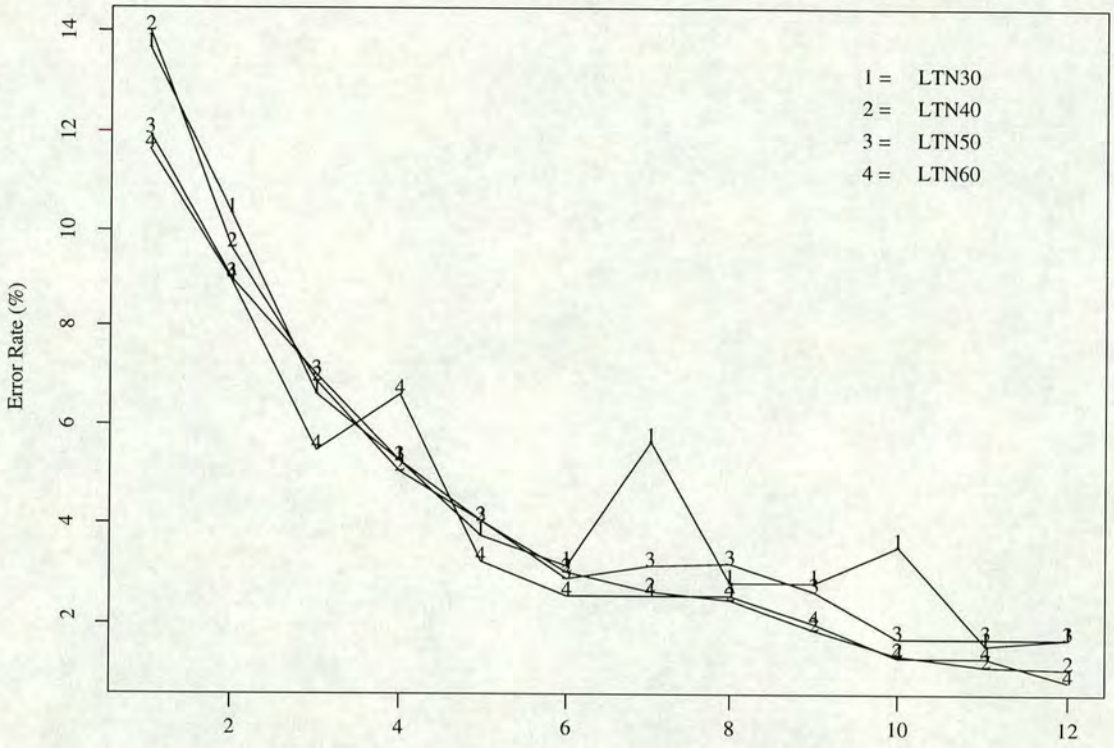


Figure 5.5: Performance over Digit Sequence Results for Four LTN Inputs.

The relative performance for different LTN values for the 12 digit sequence is given in Table 5.6. It can be seen that in all cases there are different error rates. Percentage error is the additional error obtained by using the LTN value when compared with the best LTN value. It can be seen from the table that for the 12 digit sequence LTN60 has speaker specific EER of 0.7% compared to 1.75% for LTN30. This indicates that increasing the number of inputs improves the verification performance with higher results. On the other hand an EER of 1.04% is achieved with LTN40 compared to 1.7% for LTN50. The best performance result achieved for NNM-C is LTN60. Improved performance for NNM-C with more inputs and proper alignment of the speech signals supports the hypothesis that a more detailed representation of the speech patterns proved helpful for the system. There is evidence from the results that the approaches used are well suited for the speaker verification task trained on a limited training data.

The approach of having a fixed input to the NNM-C is one of the simplest methods of

LTN	EER(%)	% error relative to BEST LTN
30	1.75	60%
40	1.04	32.6%
50	1.7	58.8%
60	0.7	0%

Table 5.6: Single LTN Set Results. Speaker Specific EERs are Given for 12 Digit Sequences.

time aligning in linear fashion of the speech signals. Although this is not the best approach to overcome the time variation of the speech signals (it does not guarantee that the internal parts of the patterns will be properly aligned) one advantage is that it does give proper alignment of the beginning and the end of the patterns. By properly selecting the time scale inputs to the NNM-C, the reduction in error rate achieved has been substantial. This method seems suitable to perform static verification of the speech signals with neural networks.

5.5.3 Digit Sequence Performance by Individual Clients

This section looks in more detail at the errors produced by different values of LTN on the 12 digit sequence using speaker specific EER. The performance of the NNM-C model can then be assessed and the best model for each of the client speakers can be determined. The breakdown of errors by client for the 12 digit sequence is shown in Table 5.7. The table shows the EERs for the different LTN values used by client speakers. The best EER over all values of LTN NNM-C is entered in the column BLTN while the worst result is shown in the column WLTN. The MEAN column has the average EER over the four LTN NNM-C for each of the clients. The best average EER is obtained with LTN60. NNM-C trained with this architecture is error free for 6/11 (54%) of the clients. From the BLTN column, if proper selection of LTN values is made for each of the client speaker then only 4(36%) of the clients have errors. LTN30 has the worst average EER. NNM-C trained with this architecture has 8 clients with errors. Table 5.7

also shows independence between the different values of LTN used and close examination of the table reveals more examples of such independence. Client 2, 9 and 10 had no errors with different LTN values used. Client 5 had errors with LTN30 and LTN40 and no errors with LTN50 and LTN60. Client 6 had no errors with only LTN40. It can be seen that the different values of LTN used in the system provide significant improvement for the different clients. Proper selection of the LTN values proved an advantage to the technique which compensate the time scale variation of the words that differ among the client speakers. In order to take advantage of this fact, perform a closed test on the training data using each of the LTN values. The LTN value that is most useful to the client can be selected and used in the NNM-C for that client. The difficulty of determining the LTN values is the limited amount of data available. Testing on the training data will not give an accurate estimate of the likely LTN. Extra data could be obtained during enrolment that could be set aside for proper selection of LTN. If the selection of the LTN values could be done without too much increase of the enrolment data, it can be beneficial to the design of the SV system. In order to improve the initial model trained on a small amount of training data, model adaptation over time will be an essential part of the SV system. The extra data obtained during this enrolment process could also be used for proper selection of LTN.

	Speaker Specific equal error rate (%)						
		L T N					
CLIENT	30	40	50	60	WLTN	BLTN	MEAN
1	0.6	0	0	0	0.6	0	0.15
2	0	0	0	0	0	0	0
3	0.6	0.6	1.2	0.6	1.2	0.6	0.75
4	4.2	1.8	5.0	1.2	5.0	1.2	3.05
5	0.6	0.6	0	0	0.6	0	0.3
6	0.6	0	1.8	2.5	2.5	0	1.22
7	0.6	0	0.6	0	0.6	0	0.3
8	8.5	5.5	6.1	1.8	8.5	1.8	5.47
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	3.6	3.0	3.6	1.2	3.6	1.2	2.85
mean	1.75	1.04	1.70	0.7	2.05	0.43	1.27

Table 5.7: Comparison of Error Rates on 12 Digit Sequences between Eleven Client Speakers with a range of LTN NNM-C.

5.6 Results using Speaker Independent Thresholds

All the EER results discussed in the previous sections are calculated using speaker specific thresholds with the SV system. In this section EER serves as a single measure of performance. A cumulative distribution of all of the client scores versus the impostor scores shows the overall degree of separation between the client and the impostor class and the results from the SV trial use a common threshold for all the speakers. The EER was evaluated for each of the 12 digits. The performance of the individual digits together with the thresholds used are listed in Table 5.8. LTN30 has an average EER of 20.0% with a range from 13% to 27%. The EER thresholds have a mean of 0.26 and vary from 0.19 to 0.37. LTN40 has an average EER of 19.5% with a range from 14% to 24%. The EER thresholds have a mean of 0.23 and vary from 0.13 to 0.31. LTN50 has an average EER of 19.7% with a range from 12% to 25%. The EER thresholds have a mean of 0.21 and vary from 0.16 to 0.27. Finally, LTN60 has an average EER of 18.7% with a range from 13% to 25%. The EER thresholds have a mean of 0.21 and vary from 0.15 to 0.30. The differences between the four LTN results are narrow although there is suggestion of an advantage with LTN60 for the 12 digit sequence results. Table 5.9 summarizes the 12 digit sequence results of all the LTN values used. The speaker verification trial with LTN30 yields an EER of 4.6% which is the worst performance for the system and 2.1% for the best result with LTN60. Use of the BLTN[2] SV NNM-C system with a speaker specific threshold has an average EER of 0.43% for a population of the eleven client speakers. This increases to an EER of 1.89% for the BLTN[3] when a common threshold was used to evaluate the system model. The percentage error gained is the gain in error when compared with the BLTN. As observed from the results, a common threshold is not a good threshold to be used with every client speaker. Previous results also support the use of client speaker specific thresholds which serve as a better estimate of the overall performance. This observation is in accordance with other researchers such as (Setlur & Jacobs, 1995)(Forsyth, 1995)(Gong, 1995).

[2]This result is taken from table 5.7

[3]The LTN parameter for each digit is speaker independent

Digit	T	LTN30 (%)	T	LTN40 (%)	T	LTN50 (%)	T	LTN60 (%)
1	0.27	18.3	0.25	14.5	0.21	14.1	0.22	15.7
2	0.22	20.4	0.20	23.0	0.16	20.6	0.20	17.9
3	0.28	21.0	0.28	22.0	0.22	20.0	0.23	18.7
4	0.23	26.8	0.21	21.0	0.18	22.9	0.20	22.2
5	0.25	15.5	0.30	14.0	0.23	17.5	0.23	17.0
6	0.21	23.9	0.20	25.0	0.17	23.0	0.18	22.6
7	0.19	24.4	0.21	22.6	0.26	24.5	0.15	25.0
8	0.30	22.5	0.19	24.0	0.20	24.7	0.30	19.7
9	0.37	14.7	0.28	12.0	0.20	15.5	0.20	14.3
zero	0.30	13.5	0.22	14.1	0.24	12.3	0.24	13.1
nought	0.21	26.5	0.13	22.4	0.18	22.0	0.20	20.7
oh	0.28	16.9	0.31	20.5	0.27	19.7	0.22	19.6
mean	0.25	20.3	0.23	19.5	0.21	19.7	0.21	18.7

Table 5.8: Speaker Verification Results using a Common Threshold (T) For all Client Speakers.

LTN	EER (%)	%error gained
BLTN	1.89	0%
60	2.10	10.0%
40	2.9	34.8%
50	2.9	34.8%
30	4.6	58.9%

Table 5.9: Speaker Verification Results using a Common Threshold (T) For all Client Speakers. EER Performance of 12 digit Sequences.

Chapter 6

NEURAL NETWORK MODELS (NNM) - CROSS MATCH TECHNIQUE WITH CLIENT BARCODE

6.1 Similarity Matching Techniques

It is generally accepted that humans can identify a person from the sound of his or her voice and yet two different voices sound alike. The distinct variation in voices has made automatic speech recognition possible while at the same time their similarities provide a challenge for speaker recognition systems (Luck, 1969). Comparisons between two speech signals are frequently made on the basis of some measure of distance between them. The simplest distance measure commonly used is the Euclidean distance. During recognition, the test utterance is compared against a reference (Rosenberg & Soong, 1986) of the claimed identity of the speaker. A predetermined threshold for the client speaker is then compared to the resulting distance. If the resulting distance falls within this threshold it will be accepted as a valid utterance for the claimed speaker. Another form of measurement is the measure of the degree of correlation or similarity between the two speech signals. For example, a speech recognition system using words spotting was developed based on the multiple similarity (MS) method for eliminating word boundary detection errors (Takebayashi *et al.*, 1991). During the recognition process, the MS values are time continuously computed for word spotting through pattern matching between an input vector and reference pattern vectors. An end point candidate t_j is assumed for each

analysis frame. Using the maximum and minimum duration (d_{max}, d_{min}), a series of start point candidates ($t_i, t_{i+1}....t_n$) are determined corresponding to the end point t_j for reference word l. The MS is applied to obtain the maximum similarity value for that word l at t_j. The process is repeated for all the vocabulary words and if the maximum similarity value is above the set threshold then the word l is spotted at t_j. It was found that the MS measure is powerful for pattern classification among words of a vocabulary but it is not totally suitable for word spotting. In a direction similar to this, a similarity technique is developed in this thesis for speaker verification systems. The focus of the work is not in word spotting but in determining intra-speaker variation based on this similarity match. Experimental results and system implementation are given in the next section to show the effectiveness of the proposed method.

In an area not related to speech, researchers have devised a grouping methodology for dendrochronology (Linton & Zainodin, 1987). Dendrochronology is a well established technique which uses the widths of the annual growth rings of trees to date timbers from buildings, waterfront structures etc. The conversion of ring widths to indices is necessary since a tree does not respond in the same manner throughout its life to a given set of environmental conditions. Individual trees, in the same way as speech tokens, respond slightly differently to the same set of conditions. In time series, such ring-width sequence is termed non-stationary. The problem of comparing many tree-ring width sequences in order to determine groups of contemporary samples is a problem that frequently arises in dendrochronology. The most useful technique for measuring the similarity between two samples is based on the cross correlation coefficient calculated between suitably transformed versions (stationary sequence) of the tree width sequences. An average sequence for each of the groups formed will be cross matched against the master chronology in an attempt to date the sequence. The above method developed by Linton and Zainodin used the largest correlation values to merge two corresponding sequences into a group.

Speaker verification - the main concern of this thesis can use correlation scores as added

information to train the neural network to classify between the client and the impostor. Speech patterns from the same client speaker should be similar when matched although speech signals of each utterance will not be exactly the same. In the work carried out here the same practical strategy of cross match was employed to evaluate local similarities within an analyzing frames of the speech signals. The aim is to compare the client speech patterns in order to group them into corresponding samples. Having found a sequence representing the group it seems sensible to attempt to form a 'barcode' representing the client. Thus the initial stage for the new approach of the SV system starts through a process of generating this barcode. The barcode is obtained from client training tokens. The training or the test utterance of the preprocessed speech signals are matched with the client barcode resulting in a correlation score. This can be easily implemented. The new score acts as added information to the existing data index (j) and the minimum distortion value (d) used as an input to train the neural network. A detailed description of developing the client barcode as well as a new approach to SV system design will be dealt with in the next section. The approach has been tested with the same database used in the previous experiments and has been shown to produce improvements in the overall performance.

6.2 Similarity Procedure for Speech Utterances

The strategy for finding the similarity of speech tokens depends on determining the correlation factor between these tokens. These tokens can be represented as vectors in space. Let X_{ij} be the vector representing the ith token of the jth speaker in the verification system. The number of reference tokens for the client speaker is represented as N. In this example if one of the tokens has 70 frames then for X_{1j} there will be 140 sample points of speech while another token X_{2j} with 60 frames has 120 sample points. If q (minimum overlap) is equal to 68 then for each pair of tokens there will be sets of 68 comparisons at different relative positions. The setting of this minimum overlapping value as well as the importance of this value is discussed in section 6.3.

In this section, the practical strategy of using a cross match technique is described by first discussing the match between two samples of speech. Two samples of speech tokens are labelled as X_{1j} and X_{2j} in which ($X_{1j} \neq X_{2j}$). The two samples of the speech tokens are compared at all possible different relative positions. In any comparison of cross match between the samples of the speech tokens there must be at least q sample points of overlap. This is to avoid the high correlation values that exist between speech tokens when matched at the beginning and end points of the speech utterances. Accordingly the number of comparisons between the two speech tokens at different relative positions is given in equation (6.1).

$$S = \frac{L_{1j} + L_{2j} - 2q + 2}{2} \tag{6.1}$$

where S = number of comparisons, L_{1j} is the length of the token X_{1j} and L_{2j} is the length of X_{2j}.

$r(X_{1j}, X_{2j}, p)$ is the score value that indicates the similarity between the two samples X_{1j} and X_{2j} when the relative position of sample X_{2j} with respect to sample X_{1j} is p(p = $-L_{2j}$ + q, $-L_{2j}$ + q +2, ...-2, 0, 2, ...L_{1j}-q-2, L_{1j}-q). If a high value of $r(X_{1j}, X_{2j}, p)$ is obtained then the two samples X_{1j} and X_{2j} are said to be very similar; a negative value shows dissimilarity between the two samples. Thus, the corresponding value of p (which will be termed as the offset) at which the maximum value of $r(X_{1j}, X_{2j}, p)$ occurs shows the best match between the two samples X_{1j} and X_{2j}. Assume that the N tokens are arranged in such a way as to form a group such that the samples within the group match each other at relative positions that indicate similarity. From the example given, tokens X_{1j} and X_{2j} are cross matched to form a group in which token X_{1j} and X_{2j} are most similar at position $p(X_{1j},X_{2j})$. By grouping similar tokens in this way, the next stage attempts to merge the two tokens within a group to form a single sequence. That is, if the similarity $r(X_{1j}, X_{2j}, p)$ between the two tokens is most similar (corresponding to a value of p at which the maximum value of $r(X_{1j}, X_{2j}, p)$ occurs) then sequences X_{1j} and X_{2j} will be

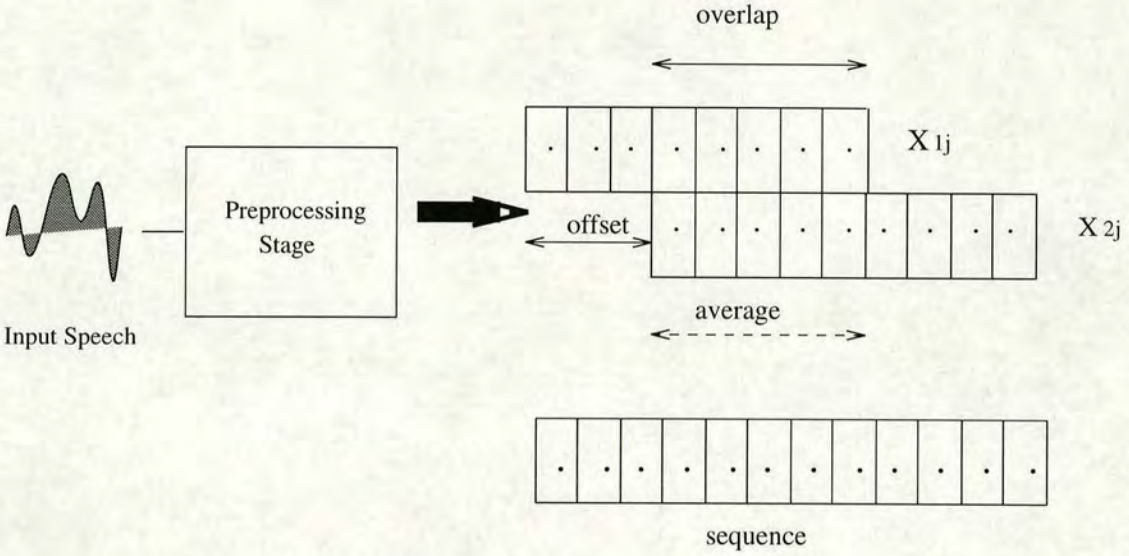


Figure 6.1: Generation of Speech Sequence from Client Tokens

merged but not otherwise. At position $p(X_{1j},X_{2j})$ which is most similar the tokens form a single sequence by making an average. The cross correlation between these tokens with the relative position of the offset is illustrated in Figure 6.1.

In this respect, the offset is a measure which shows the deviation of the time series between two samples of speech. Thus, the tokens X_{ij} refer to actual training tokens of the client speech signal. A sequence here refers to an ordered series of speech indices, composed of data from one or more tokens. After having found a sequence representing the two tokens, a further attempt is made to form a group comprising N tokens for the client by the same process. The procedure is repeated several times until all the training tokens are used up to form a group representing the client barcode.

The basic idea here is that two patterns of speech from the same client speaker of the same word should be similar when matched. The approach could assign a value reflecting the strength of the match between two sequences at each possible relative position. The statistical technique used to measure the similarity between two sequences or tokens is the correlation coefficient

calculated from the preprocessed transformed data. The correlation coefficient between token X_{1j} and X_{2j} can be represented as:

$$r(X_{1j}, X_{2j}, p) = \frac{\Sigma(X_{1j} - \overline{X_{1j}})(X_{2j} - \overline{X_{2j}})}{\sqrt{\Sigma(X_{1j} - \overline{X_{1j}})^2 \Sigma(X_{2j} - \overline{X_{2j}})^2}} \tag{6.2}$$

where $\overline{X_{1j}}$ and $\overline{X_{2j}}$ is the mean.

The r value will have a range of -1 to 1; when there is no correlation r will take a value close to 0. It is important to note that one important property of r is that it only measures the extent of the linear relationship between X_{1j} and X_{2j}. It is not helpful in detecting a more general curved relationship. In real situation the value of $r(X_{1j}, X_{2j}, p)$ calculated from equation 6.2 is not likely to be a convenient number as it may cause a rounding error (Clarke & Cooke, 1994). An alternative formula suited for calculation purposes for r is as follows:

$$r(X_{1j}, X_{2j}, p) = \frac{\sum_{k=1}^{n} X_{1j}X_{2j} - (\sum_{k=1}^{n} X_{1j})(\sum_{k=1}^{n} X_{2j})/n}{\sqrt{\left[\sum_{k=1}^{n} X_{1j}^2 - (\sum_{k=1}^{n} X_{1j})^2/n\right]\left[\sum_{k=1}^{n} X_{2j}^2 - (\sum_{k=1}^{n} X_{2j})^2/n\right]}} \tag{6.3}$$

where n is the overlap between the samples of the speech tokens.

In the work presented here, it is assumed that large values $r(X_{1j}, X_{2j}, p)$ indicate that the tokens X_{1j} and X_{2j} are very similar and that a negative value shows dissimilarity between tokens. In this case it is assumed that the large value obtained is unique to the client speaker.

6.3 Client Barcode

In order to improve further the performance of the NNM-C, a technique of developing a client barcode is introduced into the NNM SV system. In the previous section, a detailed description was given comparing two aligned speech signals to generate a speech sequence. This section is closely related to the previous section and follows the same practical strategy to generate a barcode that represents the client speaker. LTN is applied to the speech signals before the barcode generation. A more detailed procedure for generating the client barcode is given below:

Step 1: Select the N tokens, each token representing the client sample speech data for that particular digit.

Step 2 : Compare two tokens or sequences X_{1j} and X_{2j} in which $(X_{1j} \neq X_{2j})$. Calculate the correlation score or similarity value $r(X_{1j}, X_{2j}, p)$ at relative position $p = (-L_{2j} + q, -L_{2j} + q + 2, ...-2, 0, 2, ...L_{1j}-q-2, L_{1j}-q)$ using equation 6.2.

Find $p(X_{1j}, X_{2j})$ the value of p for which $r(X_{1j}, X_{2j}, p)$ is maximum. When the maximum value is found set $r(X_{1j}, X_{2j}) = r(X_{1j}, X_{2j}, p(X_{1j}, X_{2j})$. Then merge the two corresponding tokens into a sequence.

Step 3 : Calculate the values of the sequence representing the client barcode. See Figure 6.2 showing a four step process from sequence to a client barcode.

Step 4 : Get the next token. If N equals 1 then stop. Otherwise repeat step 2 and 3.

Step 5: The representation of the complete set of N tokens called the client barcode is obtained.

The strategy implemented for fixed text speaker verification is to create a client barcode that characterizes each user. Similarity scores for all the tokens are averaged across the entire speech interval to produce an overall similarity to each reference client. The primary goal of the NNM SV system is to reduce errors and most speakers normally sound very different from each other. That is to say, most client speakers are not usually rejected while impostor speakers are rarely mistaken as the client speaker. The real challenge to the problem is that the client speaker often sounds very different from time to time and this makes it a difficult task. The distance resulting from comparisons of a client's own tokens with their reference is known as intraspeaker variability. Generation of the barcode takes into account all the client tokens to become more adaptable to the intra-speaker variability. Using the cross match technique between unknown speech samples and this client barcode, the resultant similarity scores provide added information to train the classifier for verification. Before introducing the selected information (maximum correlation score) to the neural network model, this score is set to the following values:

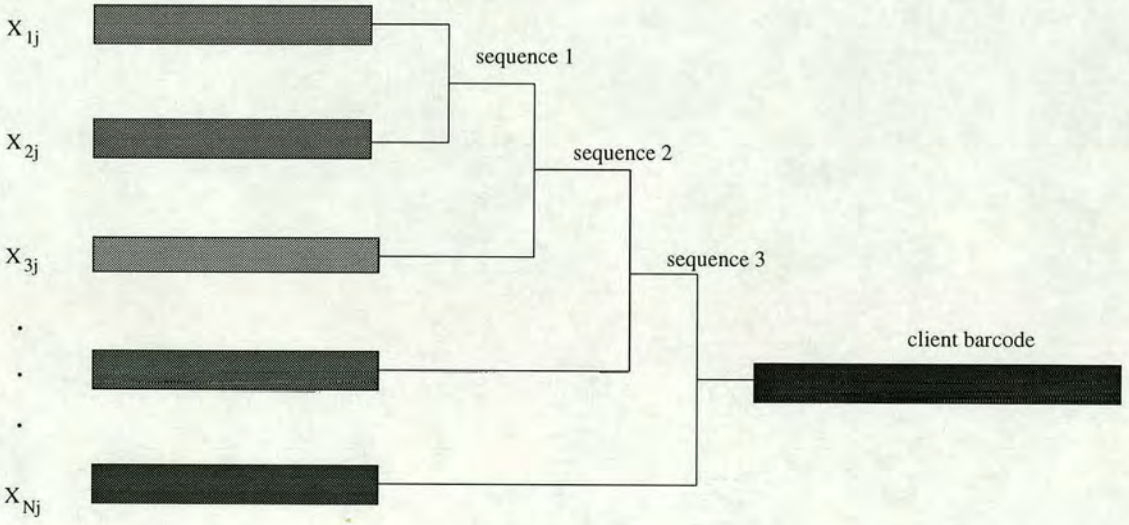


Figure 6.2: Similarity Shift with N Tokens to Generate Client Barcode

$$r(X_{1j}, X_{2j}, p) = \begin{cases} \frac{r(X_{1j}, X_{2j}, p) - X_{min}}{X_{max} - X_{min}} & \text{if } r > X_{min} \\ 0 & \text{otherwise} \end{cases} \tag{6.4}$$

Here X_{max} is set to be 1.0 while X_{min} is set to be 0.5. This normalization procedure is necessary since it will emphasize the parameter value closer to the maximum similarity.

Two important factors need to be examined before considering the practical strategy for incorporating the client barcode into the SV system and experiments to support these have been carried out. One involves the predetermination of the minimum overlap. The other is concerned with the reliability of the client barcode. In the first experiment every token was cross matched with the client barcode. There were 25 tokens taken from the client and 103 tokens from the impostor for each digit. The same procedure was carried out with the other digits and a typical example of the result is plotted in Figure 6.3. This is the distribution of the similarity scores for one token from client 1. The minimum overlapping points can be determined from this plot. The minimum overlapping points for a specific client is the averaged value from all the digits use in the experiment. On the left hand side of the figure, for every comparison made there is an increment of 2 points of the sample speech overlap while the left hand side shows decrement of 2 points of the sample speech. As can be observed from the figure, selecting the minimum overlapping points is important. The longer the value of overlapping points at the center the better is the result. If the number of overlapping points is too small as indicated on the left and right hand side of the figure, the results obtained are unstable. It would be an advantage to set a minimum ovelapping points in order to avoid such case. When two samples are very similar the results show a high similarity value. The same procedure was carried out with four other speakers and the result is given in Table 6.1. The average minimum overlapping points obtained from these speakers is 68. This is the minimum overlapping value (q) set. This value is used

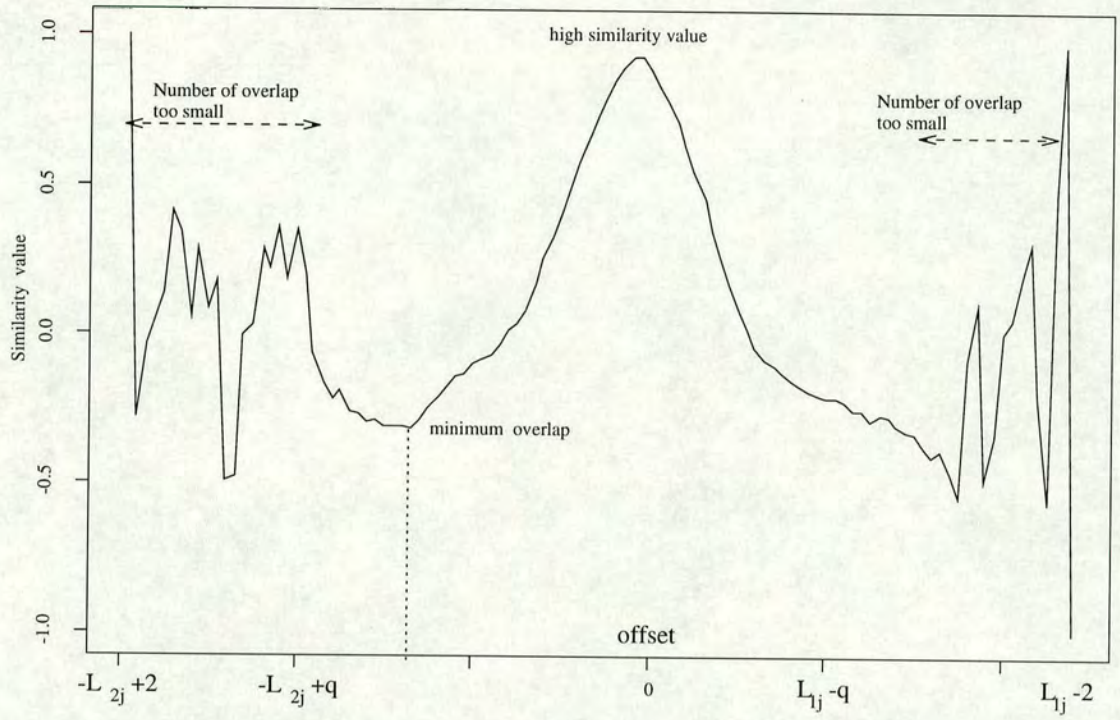


Figure 6.3: Distribution of Similarity Scores Versus Offset For Client 1.

with the other remaining client speakers in the remaining experiments.

In the next experiment, the reliability of this client barcode was plotted to examine if the similarity score gives a measure of the quality of the match. All the training and the test tokens from both client and impostor speakers were cross matched with this barcode. Figure 6.4 and Figure 6.5 are plots from the distribution of the similarity score between the client and impostors for a specific speaker and digit. The same number of tokens as were used to determine the minimum

Client	1	2	3	4	5
q	66	66	76	66	70
mean(q)	68				

Table 6.1: Minimum Overlapping (q) Sample Points for Client Speakers.

overlap were also used in this experiment. The high similarity scores are distributed to the right hand side of the figure: an indication of tokens belonging to the client while the low values indicate otherwise. One important property of the neural network which is useful to the ASV task is the ability to represent the statistical properties of data. This does not mean that all forms of input representation are equally effective in the neural network. Since the initial result of having this barcode is promising, this permits the NNM SV design to focus on the construction of optimal input features. The next section will build on the success of the preliminary study by incorporating a cross match technique into the NNM SV system framework.

6.4 NNM-CM SV System

This section is concerned with describing the neural network model trained with the additional source of information gathered from the cross match technique known as NNM-CM (Neural Network Model - Cross Match system). The new ASV system is designed to further improve the discriminative ability of the binary classifier and its performance on the ASV task is assessed. The speech samples from either the client or the impostor are cross matched with the client barcode to obtain the correlation score as well as the offset value at a relative position that indicates the highest similarity between the speech samples. These new features are added to the output of the preprocessor which contains the codevector (j) and the minimum distortion value (d). These new representations of the input features are passed to the classifier which is the MLP as used in the previous chapter. This section begins with an explanation of the importance of the new input representation which enables the NNM-CM SV system to achieve the prescribed task. This is followed by discription of the principal operations of the whole system and finally by the motivation in the development of the new approach.

In the previous NNM SV system, the object was to accept or reject the identity claim of a client speaker by the use of frame labelling from the client codebook known as the NNM-C.

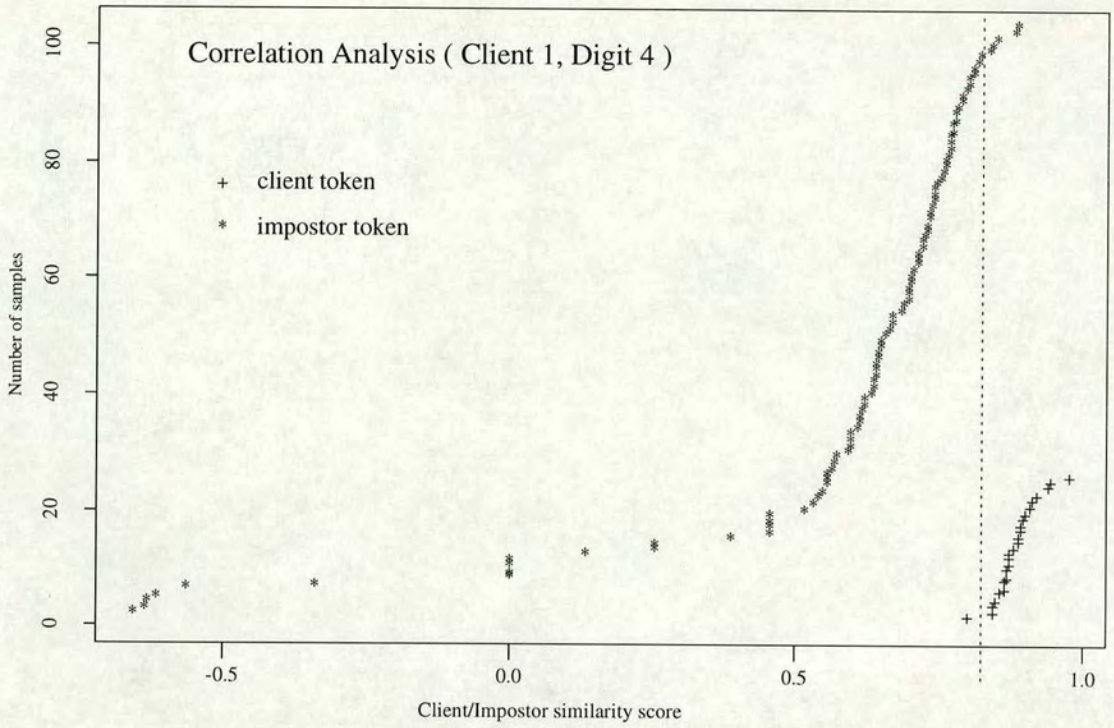


Figure 6.4: Correlation Analysis: Client 1 for Digit 4

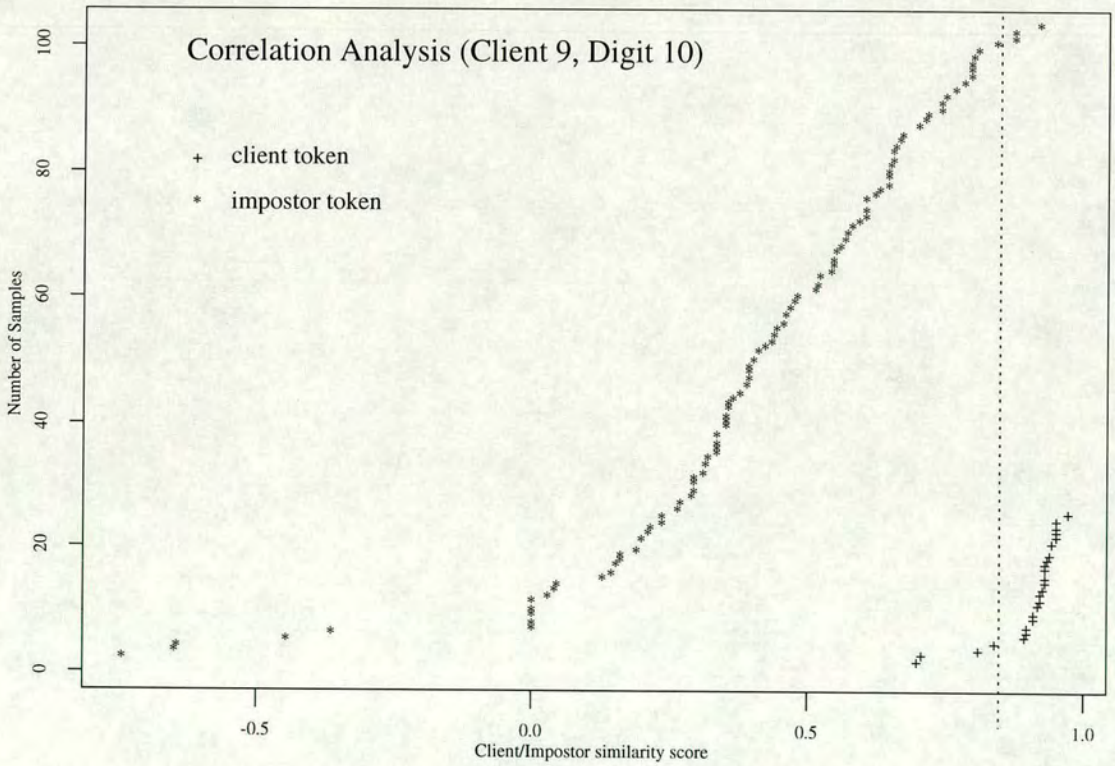


Figure 6.5: Correlation Analysis: Client 9 for Digit 10

This model was trained without the new correlation scores and the offset value which reflect the degree of similarity and the relative timing position to the rest of the client population. The new form of input representation emphasized the similarity/dissimilarity between the client and the impostor data and this will have a different influence on the number of hidden nodes required, the iteration as well as its accuracy on the training and the test data as the network learns the required mapping. A better representation of the inputs can encode useful information about the domain in a manner which will aid the learning process. The addition of the new input representation to the SV system can produce outputs which better model the client characteristics, thus giving better performance results in a superior SV model than the other NNM SV system. To test this hypothesis two experiments were carried out and the detailed results are given in the next section. In the first experiment, the complete input representation of the NNM-CM consisted of index (j) and the minimal distortion value (d) plus the similarity score. Experiment 2 makes use of the same architectural input with the addition of the offset value. The primary goal of these experiments is to evaluate the new technique that combines the desirable characterization of r and offset.

A block diagram indicating the principal operations of the system is shown in Figure 6.6. Basically the classification of a speech utterance using a neural network SV system consists of a feature extraction stage, preprocessing stage, classifier and the decision logic. The sample utterance is fed into the system where the signal is end point detected. The utterance was then analyzed for feature extraction of the cepstrum coefficients. The preprocessing stage as well as the database used for the NNM-CM parallel those of chapter 5. The barcode obtained as discussed in section 6.3, was then cross matched with the preprocessed utterance to obtain the similarity or dissimilarity scores corresponding to the client or the impostor utterance. For NNM-CM with LTN60, two coefficients per frame plus the similarity score will fit the 121 input units. The neural network design was basically the same as that of the previous chapter. The training algorithm proceeded in exactly the same fashion as in the previous experiments with

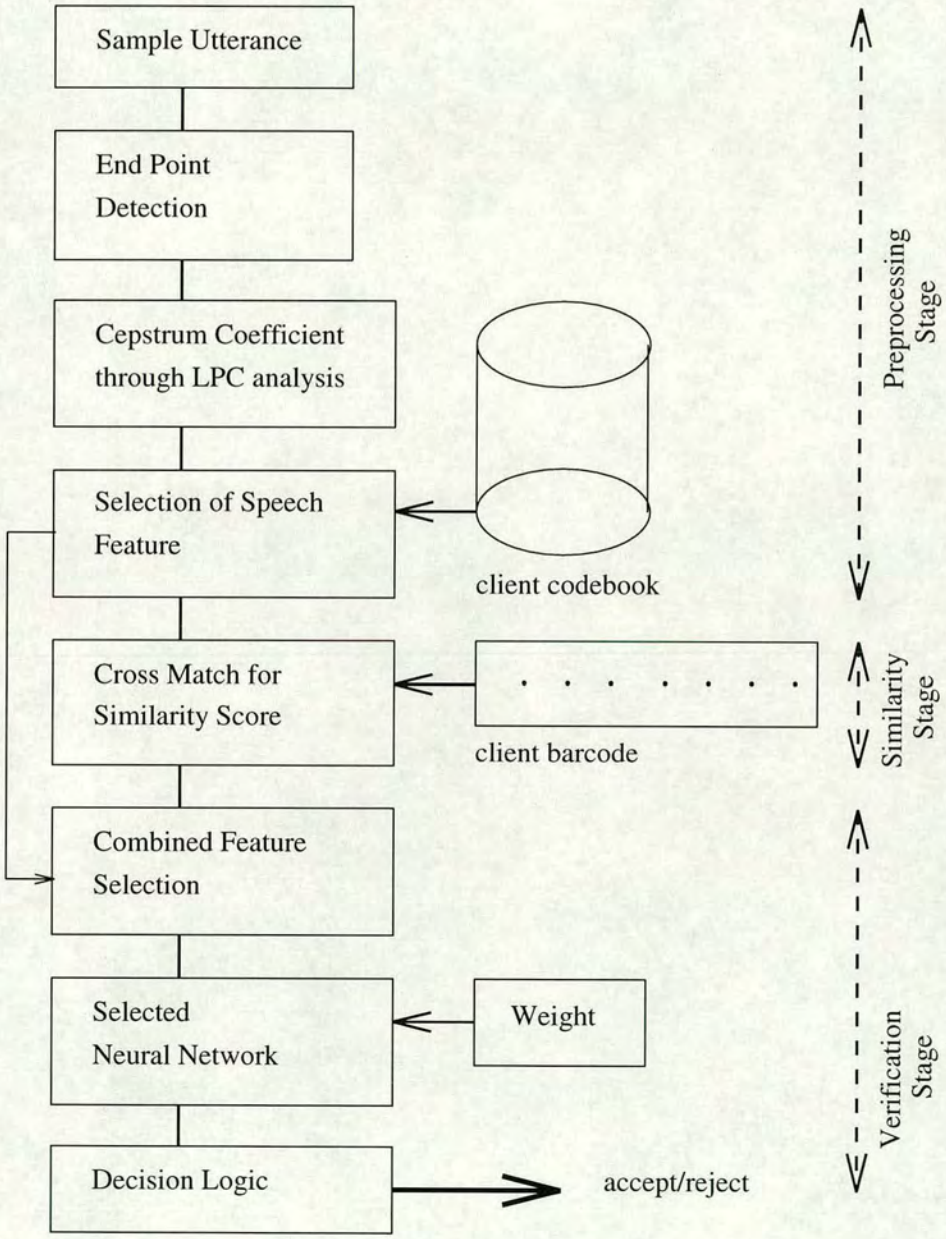


Figure 6.6: Automatic Speaker Verification Operation. NNM - CM SV System

the gain and momentum term selected according to the results of the preliminary experiments. The neural networks were repeatedly trained and the final weights recorded for the system. The same random seeds were used for each of the training sessions. In the final stage, the classifier decides whether the given utterance pattern belongs to the given category or not. The output of the neural network gives a similarity measure between the given utterance input pattern and the category for which the neural network was trained. It will be shown later in the section that the number of hidden units required for generalization decreases as the number of inputs decreases for different techniques of NNM used in the system. As in the previous experiments, the results clearly show that the neural network learns the new mapping with improved accuracy.

The motivations for development of the new SV system include the interest in experimenting with different network architectures as well as explicitly incorporating new information whilst maintaining past information into the architecture which can result in stability and convergence of the network design. Some other technical benefits of the NNM-CM include:

- The development of the client barcode can be derived from the NNM-C preprocessing stage with minimal extra processing.
- Additional information sources available for training the neural network.
- Ease of implementation and ease of incorporation into the existing NNM SV system.
- The results shows the NNM-CM SV system provides a modest performance improvement over the NNM-C system.

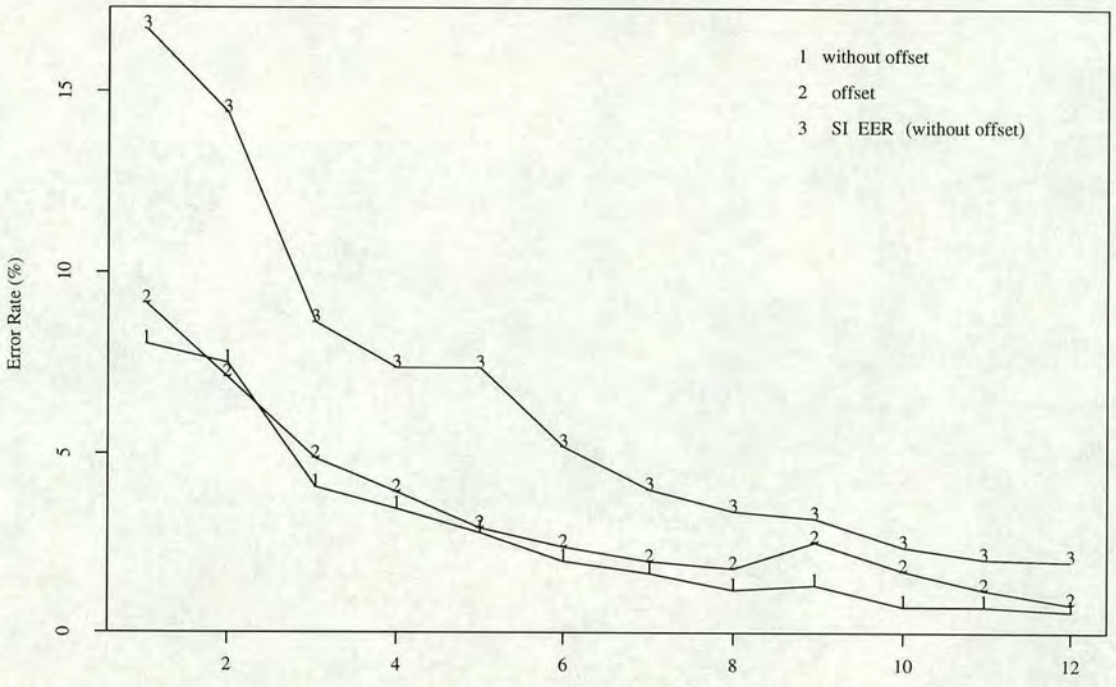


Figure 6.7: EER for Each 12 Digit Sequence Length using Speaker Specific and Speaker Independent Thresholds. NNM-CM SV System Trained with and without Offset.

6.5 System Performance

In this section, the potential of NNM-CM for ASV is explored directly using experimental procedures mentioned earlier in the chapter. The normal practice reported in the literature for measurement of system performance is the EER. However, there are other approaches to measure system performance which will be considered at in the next chapter.

The speaker acceptance or rejection decision in this section is carried out by comparing the results of the new approach to the previous work of the NNM SV systems described in chapter 5. Both speaker specific (SS) and speaker independent (SI) thresholds were used to evaluate the SV system of chapter 4 and chapter 5. These thresholds are determined by the EER criterion. The use of EER in both chapters provides a standard set of measurements that detail the performance of SV systems. The first step is the evaluation of performance on a single isolated digit test utterance. In this experiment the normalization procedure made use of LTN60 for the evaluation of the NNM-CM SV system. The EER was evaluated for each of the 12 digits. The performance

of each of the digits and the threshold used are listed in Table 6.2. For a speaker specific (SS) threshold the average EER is 11.8% with a range of 8% to 15% while for a speaker independent (SI) threshold, EER has an average of 18.7% with a range of 13% to 23%. The SI thresholds have a mean of 0.339 with a range of 0.183 to 0.547. The graphical representation of the EER for various digit sequence lengths is shown in Figure 6.7. The figure shows the performance improvement of SS threshold over the SI threshold for the various digit sequence lengths. It can be seen that there are large differences among the digits in the distribution of errors as noted in this chapter as well as the previous chapter for NNM SV systems for single digit performance. Also in general the relative performance of the digit sequence shown in Figure 6.7 follows the same pattern over the other NNM SV systems. The performance results of NNM SV systems are likely to be affected by a number of factors. Firstly, the use of multiple information sources obtained from different types of codebooks. Secondly, the size and network complexity are under the constraint of limited training data. Thirdly, the capability of the preprocessor in handling the temporal structure of the speech signals. Finally, the input representation to the neural network varies and this is dependent on the type of preprocessor used for the SV system. These preprocessors contain vital information of the speaker characteristics.

In the next experiment the performance ofthe NNM-CM with addition of offset data is evaluated. It is noted from the results that addition of offset information provides no significant improvement in EER. The extra information provided is not necessarily better as seen in Table 6.2 and Figure 6.7. An explanation is offered for this. Firstly, the amount of data available to develop the client barcode is limited, thus it is unable to cope with the changes of the client speaker characteristics. More importantly the grouping of the client tokens to form the barcode may be less robust to determine the best relative position which determined the deviation between two samples when matched. For example, probably a much better approach is by (Linton & Zainodin, 1987) which used the form

$$p(i, k) = p(i, j) + p(j, k) \tag{6.5}$$

to satisfy the relative positions that are consistent with each other to form a group. The above equation means that sample i and j are most similar at position p(i,j) and that j and k are most similar at position p(j,k). The samples that form the group are combined provided that p(i,j) + p(j,k) gives the position at which i and k are most similar that is p(i,k). In their approach, all samples are considered at the same time to form a single sequence representing the group. When an unknown sample is matched with the group, not only will it have a high correlation score but it also gives the best possible, relative position (offset) that is consistent to each other.

Digit	SS EER (%)		SI EER (%)	
	No offset	offset	No offset	threshold
1	8.8	9.4	16.8	0.325
2	13.4	14.1	21.2	0.183
3	11.0	12.1	18.7	0.254
4	13.8	14.7	23.4	0.290
5	9.3	10.0	14.9	0.400
6	14.7	18.0	18.4	0.293
7	15.3	16.1	22.7	0.193
8	12.2	13.0	18.6	0.408
9	9.8	10.3	14.2	0.334
zero	8.4	9.2	13.5	0.486
nought	15.3	16.2	21.1	0.359
oh	10.3	11.0	21.6	0.547
mean	11.8	12.8	18.7	0.339

Table 6.2: EER and Threshold Digits Using Speaker Specific and Speaker Independent Thresholds. NNM-CM SV System Trained with and without Offset.

6.5.1 Improved Design of NNM SV System

This section examines how the NNM-CM compares to the NNM-C with LTN60. For the single digit SS EER performance NNM-CM has an average of 11.8% compared to 11.2% and 17.8% of the NNM-C and NNM-CI respectively. There is not much difference in the overall performance between these two models (NNM-CM and NNM-C) for the single digit result. Even though the EER for the NNM-CM system is slightly higher for the single digit than the NNM-C system, the EER NNM-CM appears to perform better for the 12 digit sequence results. The NNM-C produce an EER of 0.7% (SS EER) and 2.1% (SI EER) on the sequence of 12 isolated digits. The NNM-CM produce an EER of 0.62% (SS EER) and 1.9%(SI EER) on the sequence of 12 isolated digits. The SS and SI EER result is improved slightly which represents a modest 11.4% and 9.5% improvement over the NNM-C. Figure 6.7 also shows the comparison of the NNM CM system performance with different types of threshold.

Differences among speakers in their performance on SV systems have been reported by many studies (Forsyth, 1995)(Rooney, 1990)(Doddington, 1985)(Tsoi *et al.*, 1994). The results reported here confirmed those reported above. Figure 6.8 shows the differences in performance of the EER between the conventional NNM-C and the NNM-CM for the 12 digit sequence EER for each client speaker. This chart can be an effective tool to make a quantitative judgement between NNM SV systems. It also addresses the questions relating to the effects on client speakers. This allows judgement to be made on how the various client speakers differ in percentages used for the two NNM SV techniques. The c represents the client trained from the conventional NNM while the cm is from client trained with the NNM-CM.

For client speakers 7, 8, 10 and 11 no errors were reported using either of the two models. Client speakers 1, 3 and 6 with NNM-CM offer a significant improvement in performance with no errors compared to the conventional model; client 4 also had better performance with NNM-CM. Client 5 shows no significant difference between the two models while client speaker 2 and 9 of the conventional model produce fewer errors. It is interesting to note that the inclusion

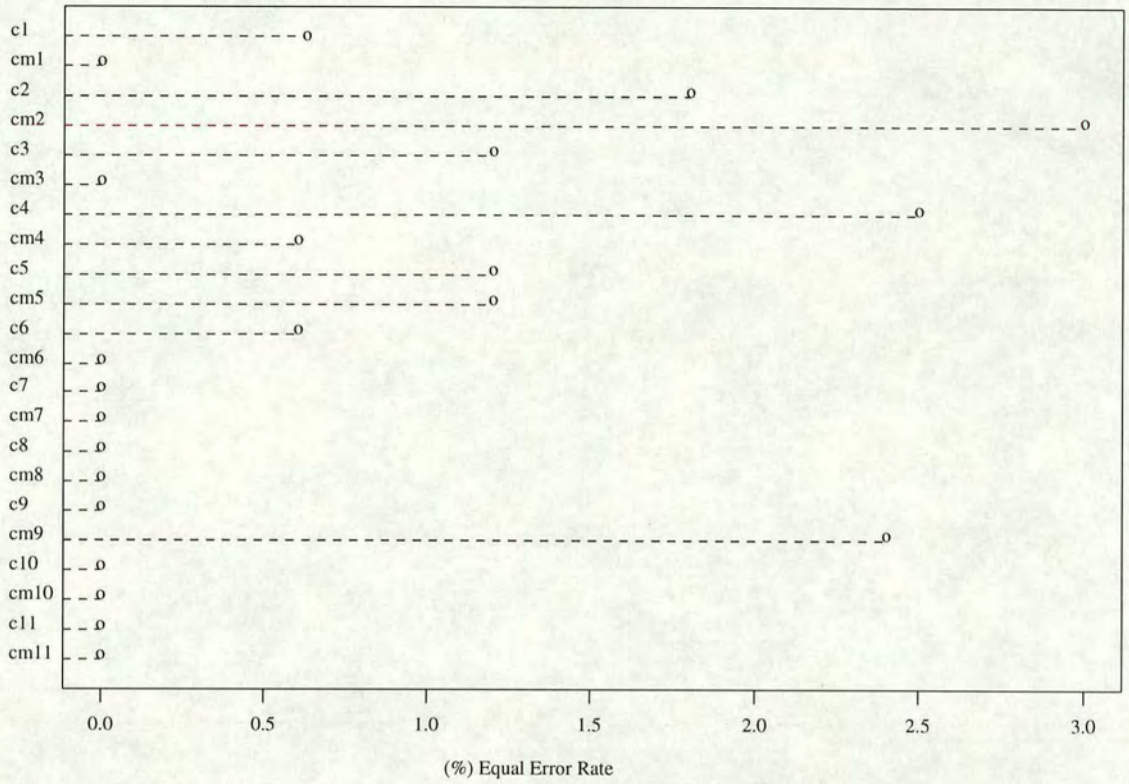


Figure 6.8: Client Distribution Data with SS EER for 12 Digit Sequence Lengths

of the similarity score helps stabilize thresholds only for certain speakers. For these speakers the added information helps the classifier to form a better decision boundary to produce better EER performance. NNM-CM trained with this architecture is error free for 7/11(63%) while the NNM-C is error free for 6/11(54%) of the clients. Note that each of the two techniques has advantages and disadvantages for the selected speakers.

It can be seen that there are large differences amongst speakers in the distribution of errors and these errors are not uniformly distributed which indicates that the NNM SV systems will vary significantly between speakers. The worst two clients are client 2 and 4 for the NNM-C and 2 and 9 for the NNM-CM. The middle category of client performance is represented by the other remaining clients.

6.5.2 Threshold Settings and System Adaptation

The previous sections have addressed questions relating to the effects of the speakers themselves as well as system performance. The EER values reported above show the performance of the SV systems with speaker specific (SS) and speaker independent (SI) thresholds. The use of this EER implies a perfect threshold, which is not possible in real applications since the threshold has to be determined *a priori*. Figure 6.7, shows that speaker specific distance thresholds have advantages over SI thresholds. An SV system which lacks the use of SS threshold will have an increase in EER. Thus, the use of SS thresholds can overcome speaker variability. However, the SS threshold cannot be reliable in this case as this threshold is calculated from the small amount of data available for this specific application. This means that the threshold used may be less robust than the SI threshold. The use of a different threshold for each digit and for each speaker will definitely improve the performance of SV system but in real applications where training data is restricted, the use of a SS threshold is not encouraging. The solution to this problem is to obtain more data in order to reliably estimate the threshold. This could be implemented during the verification process for the SV system. With this additional data a reliable estimate of the distribution of the open test true client speaker is possible. In situations where the client is replacing an existing service with a more convenient one, then the success of this new service (eg mass-market telephone banking) will be largely dependent on the inconvenience to the user. Clients may not be willing to record more than five tokens (Forsyth & Jack, 1993).

In addition to the problem of having limited data to reliably estimate the threshold, system performance normally degrades over time. This is illustrated as drift in Figure 6.9. As reported in (Doddington, 1985) a speaker's attitude to the use of SV system changes over time as their experience over the system grows. This means that at the initial stage the error rate is high but gradually decreases as users become familiarised with the system. With reference to the same figure, this is shown as familiarisation and drift. Another contributing factor is that the client speakers vary their speaking rate differently and non-uniformly which affects their voice

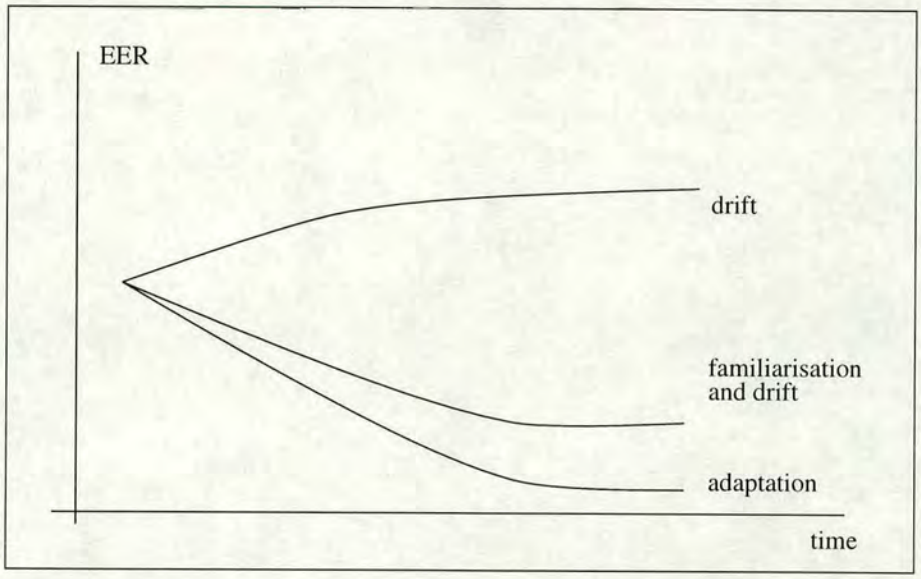


Figure 6.9: Improvement of System Performance over Time Through Adaptation and Familiarity.

characteristics from time to time. These factors mean that in order to obtain a reliable threshold and steady performance of a system there is a need for adaptation. There is evidence from literature that adaptation can improve the performance of the SV as well as speech recognition system (McInnes, 1988)(Rooney, 1990)(Rosenberg *et al.*, 1990). It is suggested that adaptation can be implemented for NNM SV systems in the following manner. During the use of the system by the client, the successful bid is updated into the system; the score is used to adapt the SI thresholds to make them more speaker specific. This means that the performance of the SV system will improve over time. By adaptation, the threshold used will be constantly revised, as the system's knowledge of intra-speaker and inter-speaker distance distribution increases. At the same time, information from a successful bid is included in the reference by cross matching with the client barcode, so that the client barcode gradually accommodates to any changes in the speaker's characteristics. Adaptation may therefore provide a solution both to the problems of changes in client speaker performance over the longer term and to the need for large amounts of training data for reliable threshold estimates. It is predicted that over a longer term NNM-CM SV systems will have better improvements in performance than NNM-C SV systems.

Chapter 7

COMPARISON OF TECHNIQUES FOR SUCCESSFUL APPLICATION OF NEURAL NETWORKS

7.1 Performance Measure Strategies

Performance measurements are often difficult to provide for any biometric system but are important criteria for the evaluation of the system performance. In the case of speaker verification the main difficulty involves the expense and effort in collecting and testing a statistically representative speaker database. The effectiveness of particular systems is best described in terms of their error rates. The receiver operating characteristics (ROC) which plot the correct acceptance against false acceptance on the vertical and horizontal axes respectively with varying threshold is also used by researchers (Hangai *et al.*, 1992)(Furui, 1994)(Reynolds & Carlson, 1995) in evaluating the SV system performance. The ROC curves give a good representation of the trade-off between FA and FR. They can also be an important tool when used with neural networks because the results obtained are not sensitive to the probability distribution of the training/test set data. Details on the ROC as a way to measure the performance of the neural network can be found in (Eberhart & Dobbins, 1990). Another alternative to the EER is the minimum error rate (MER). This is the point where the sum of the FA and FR reaches a minimum. The MER approach is less useful as an indicator of system performance since it is influenced heavily by the

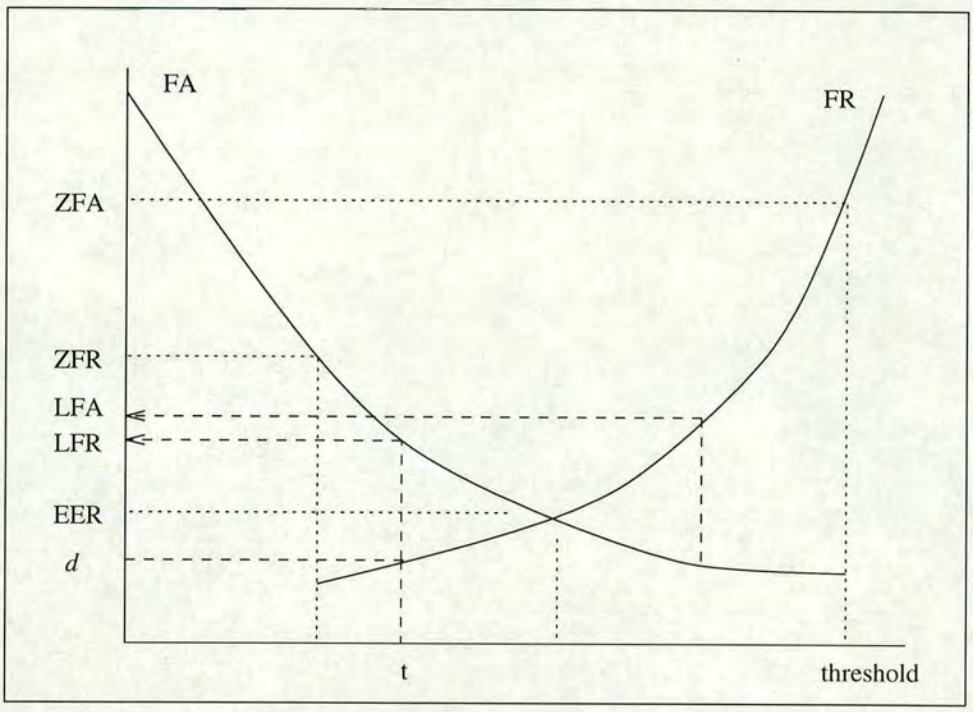


Figure 7.1: Method of Setting Speaker Verification Threshold to Avoid Outliers in Test Data.

shape of the distribution of inter-speaker and intra-speaker distances (Rooney, 1990). In another related study, performance was measured by the number of errors expressed as a percentage of the total number of verification attempts.

The EER is the most common performance measure used throughout the work in this thesis. This is simply because of its common usage by other researchers in the literature. However, another performance measure that analyzes the separation of the true scores and the impostor scores is still required to compare various techniques and system reliability. The other performance measures that can be used to evaluate the SV system are the zero false rejection (ZFR) and the zero false acceptance (ZFA) which measure different aspects of performance. The ZFA is the false rejection rate with the threshold being set so that there are no false acceptance errors. The ZFR is the false acceptance rate with no client speaker being rejected. The absolute EER is of limited interest here because applications are so task dependent. In applications such as

telephone banking, the ZFR performance measure can be more relevant to the acceptability of the NNM SV system. If security is the key issue then the ZFA is the more appropriate measure of system performance. However, care must be taken when using ZFA and ZFR as performance measures as they can be unstable[1]. ZFR depends on the single worst-matching client utterance while ZFA depends on the best-matching impostor utterance. In practical applications it would be better to adopt a small non-zero FR (d), and find the corresponding FA rate, instead of relying on the ZFR. Here, t is the threshold which determines the LFR (low false rejection). It can be seen in Figure 7.1, the value of t for the LFR is slightly higher than for the ZFR. The same principle can be applied to ZFA performance measure.

7.2 Comparison with HMM Techniques

The performance of neural networks and HMM varies strongly with the amount of training tokens available. The number of training tokens needed varies according to the network structure and the input data. It seems that the larger and more complex the input space of a particular pattern the more training tokens will be required (Maren *et al.*, 1990). As noted before in speaker verification applications for telephone speech it is important to be able to construct models with the least amount of data. Five training tokens are commonly used (Rosenberg *et al.*, 1990), based on the minimum amount of data required to train a reliable HMM SV system.

When comparing the work carried out in this thesis with other published results it is important to ensure that different systems are trained on comparable amounts of training data and that the performances of the systems are evaluated on the same database. Comparisons of different systems which use different amount of training data and different databases are not very meaningful. In view of this, a comparison was made with the established technique of HMM (Forsyth

[1]The ZFA and ZFR are highly sensitive to outliers in the test data.

et al., 1993) applied to the same database. The database used to evaluate the NNMs is similar in size and design. For DHMM (Discrete Hidden Markov Models), models with 3 and with 6 states were constructed for each digit. A SCHMM (Semi Continuous Hidden Markov Model) with 6 states was constructed for each digit. Both of these models were trained with 5 and 10 training tokens and tested with 10 true client tokens and 95 or 100 impostor tokens for each digit for each speaker. The NNM-CI and NNM-C SV on the other hand were trained with 5 client tokens and 19 impostor tokens and tested on 20 true client tokens and 83 impostor tokens. EER for individual digits for DHMM ranged from 12%-28% for the 5 token models and 8%-17% for the 10 token models. Average EER of 14% (DHMM) and 12% (SCHMM) were achieved for single isolated digits and 4%(DHMM) and 2% (SCHMM) for a sequence of 12 isolated digits for the 10 token models. NNM-C SV experiments, with 40LTN, produce an EER of 1.04% compared to 4% for DHMM trained with 10 tokens. The NNM-CI produce an EER of 3% on the sequence of 12 isolated digits. This favours the neural network approach considering the differences in the amount of training data. NNM-C produced an average EER of 13% compared to 18.5% (DHMM) and 14.3% (SCHMM) for single isolated digits. One difference in Forsyth's system from the NNM is that a standard codebook is used for all speakers and for all digits instead of different codebooks for all speakers and digits. Another important difference is that HMM just models the client data whereas NNM is trained to discriminate between client and impostor data. Some indication of the effect of different systems for speaker verification tasks can be gained from the results. As mentioned earlier the NNM-C for the 12 digit sequence using 5 training tokens per digit produced an EER of 1.04%. This compares favourably with the 2% EER of Forsyth's system.

In (Rosenberg *et al.*, 1990) a continuous hidden Markov model (CHMM) trained with 8 tokens using a database collected in a quiet environment provided an EER of 1.1%. Apart from differences in the amount of training data, the database used is free from background noise

as well as variations in the telephone handsets. One possible reason that accounts for good performance of the NNM-C is that having digit specific codebooks may improve the robustness of the model.

Despite these results the full benefits of the neural network approach have not yet been utilized for verification. Results have shown that selecting the best LTN from different digits resulted in better performance on sequence of the 12 digits. The usefulness of this approach is investigated further in the next section.

7.3 Comparison with Discriminative Observation Probability HMM Technique

Two different speaker verification methods were described in the previous sections. All of them are based on neural networks; the first one used a preprocessing stage with the client codebook alone while the second one used the client and the impostor codebook. Both of these methods are compared with the traditional method of HMM for the speaker verification task. While the baseline HMM performance is poor (mainly due to lack of explicit discrimination between classes), techniques such as discriminative training (Naik *et al.*, 1989) (Chou *et al.*, 1993) (Li *et al.*, 1995)) and cohort normalization (Rosenberg *et al.*, 1992) (Higgins *et al.*, 1993) have been shown to improve performance. In another approach a discriminating architecture known as discriminating observation probability (DOP) HMM also has been shown to improve performance with limited training data of telephone speech (Forsyth, 1995). The DOP model involves constructing a discriminating model from two standard models (one for client and another for the general population) without the need for discriminative training. The DOP technique has successfully improved the performance by combining several feature sets. The four feature sets used are the LPC cepstra, delta cepstra, mel-frequency cepstral coefficients (MFCC)

DOP HMM	NNM-C
* Multiple features provide the system with multiple information streams from which the verification decision can be made. Combining of multiple information streams from models based on different features sets produces significant benefit because of the independent speaker discriminating information they contain.	* Only single feature set used in the model
* The system is capable of modelling the temporal structure of the speech signal and this has been the great advantage of the HMM model.	* MLP preprocessor has a fixed input size which will not effectively deal with the time varying nature of the speech signal.

Table 7.1: Advantages of DOP HMM over NNM-C

and difference MFCC. The combination of the information streams used a simple linear weighted sum. The same number of training tokens is used for both the NNM and the HMM system. The DOP HMM is evaluated with 21 client and 80 impostor speakers. Comparison of fixed text verification results with other systems can be difficult because the different systems might use different types of password, decision strategies, speech databases and different amounts of training and test data. The systems under comparison must be evaluated with speech databases that reflect the operating conditions of the telephone network. Comparison of the methods used in this thesis is made with the established technique of HMM and the differences in the amount of training and test data between the two systems are given in the previous section. The optimal HMM DOP technique with cepstra and MFCC provides an EER of 0.17% and an equal EER performance result of 0.79% using cepstra features compared to 0.7% by NNM-C with 60LTN. The DOP technique is superior in performance compared to the approach used in this thesis. Table 7.1 shows the main reasons for the improved performance by DOP HMM over the NNM-C baseline system.

The DOP HMM ASV system has a powerful discriminative ability but the drawback of the system is that training the system can be complex as it requires extra preprocessing. A

conventional HMM has to be trained for the client model followed by the reference model. It also involves taking the differences in the observation probabilities between these models and normalizing the differences into probabilities in the range 0 to 1. Using these probabilities creates the DOP HMM model of the client speaker. Training the NNM-C is much simpler than for the DOP HMM SV model. The discriminative ability of the NNM-C as one model is the main advantage as it is more efficient than the discriminating models of HMMs.

Since HMM has an efficient training and testing algorithm, HMM can be an added advantage as a preprocessor. The HMM preprocessor can prepare the classes to be distinguished from each other. An SV method which combines an HMM based preprocessor with the neural network architecture was implemented and detailed information is presented for the combination of these models in chapter 4. System comparison of this approach with other techiques will also be discussed in the next section.

7.4 Discussion of Results for Different Techniques

This section compares the performance measures used to evaluate different SV techniques. The results are presented and discussed. Comparison with different techniques of SV systems in the previous sections used SS threshold to calculate the EER results. The relative performance of the different techniques for the 12 digit sequence is given in Figure 7.2. In this remaining section the ZFR and the EER threshold used are digit specific but speaker independent.

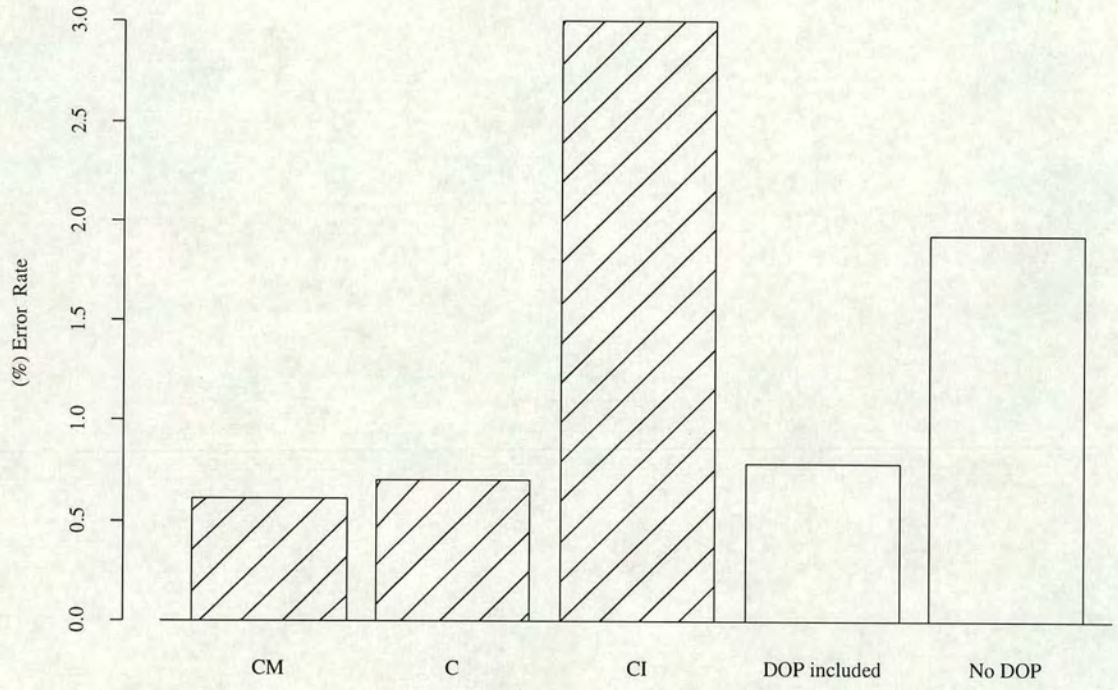


Figure 7.2: Comparison of Percentage Equal Error Rates for 12 Digit Sequences with Speaker Specific Thresholds for Different Techniques Used in SV. a) CM: NNM-CM b) C: NNM-C c) CI: NNM-CI d) DOP included: SCHMM trained with DOP e) No DOP : SCHMM trained without the DOP

7.4.1 Speaker Independent EER Performance Evaluation

In line with the objective of this thesis, Figure 7.3 illustrates the comparative EER performance with all the applications of neural network preprocessing techniques used in SV systems. The analysis given here compares the four different techniques used in the design of the SV system. There is an increase in performance as measured by EER when the similarity score is added to the NNM SV system. The result is an indication of the usefulness of this added information to the system in general. The similarity score analyzes the relationship between high correlation of the true tokens and lower correlation from the impostors and this is an added advantage to the approach used for the NNM SV system. Furthermore the complexity of the NNM-CM system is roughly equivalent to that of the NNM-C. The above reported results clearly demonstrate the viability of the NNM-CM based neural network SV system. The use of NNM-CM SV system shows an improvement of 9.5% compared to NNM-C and 61% to the NNM-CI SV system. NNM-CM is superior to the NNM-C but needs to be verify on a larger set of speakers. The EER performance measures support two basic conclusions:

- NNM-CM based systems learn representations that are qualitatively different from those learned by the NNM-C based system.
- Input representations for the NNM-CM are more appropriate (for a given ASV task) than the NNM-C or NNM-CI based system.

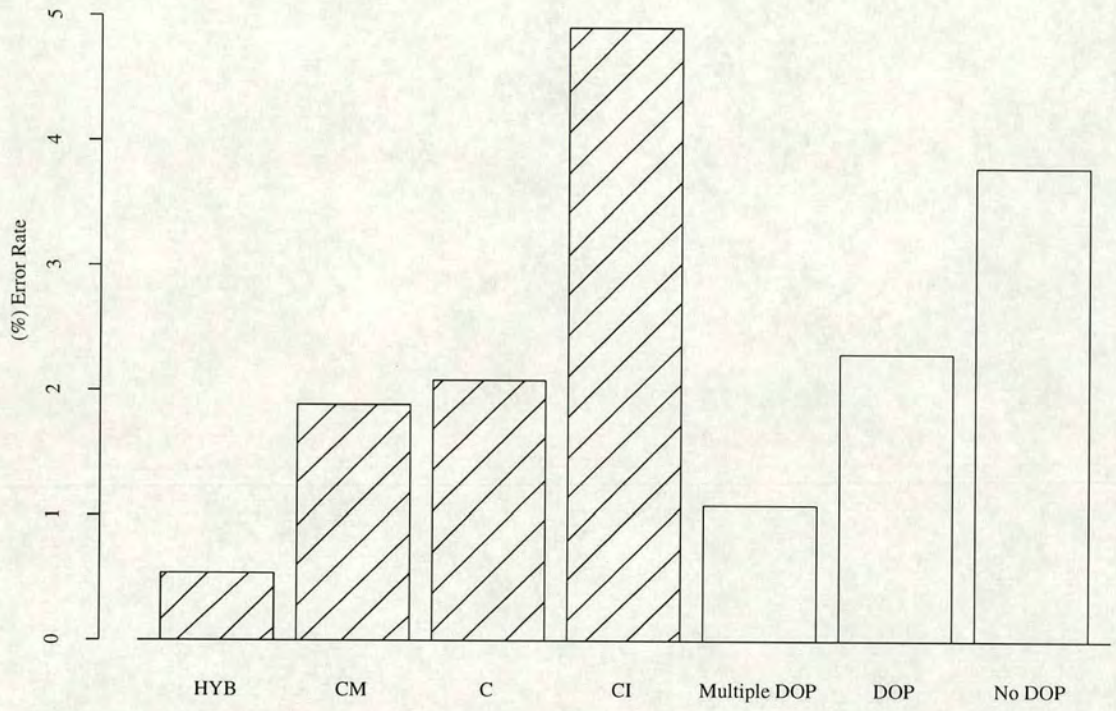


Figure 7.3: Seven Way Comparison of Percentage Equal Error Rate on 12 Digit Sequences with Speaker Independent Thresholds for Different Techniques Used in SV. a) HYB: HMM-MLP b) CM: NNM-CM c) C: NNM-C d) CI: NNM-CI e) Multiple DOP : SCHMM trained with DOP cepstra plus DOP delta cepstra f) DOP : SCHMM trained with the DOP cepstra g) No DOP : SCHMM trained without the DOP

As reported earlier in chapter 4, section 4.1.2, there is experimental evidence that applying an HMM preprocessor will enhance the discriminative ability of the NNM classifier. When compared with the other SV systems reported in this thesis the best performance achieved is with the HMM-MLP SV system. It performs better than the conventional NNM-MLP systems. The information gathered in all the experiments illustrates the problem encountered when trying to optimize a SV system using limited training data. In the case of the NNM-CI SV system the network size for this system is double compared to other NNM SV systems. This is because the inputs to the neural network come from the client and impostor codebooks. All NNM SV systems are trained with the same amount of data tokens. On the other hand the reduction in error rate achieved with the NNM SV system trained under the same conditions with only the client codebook has been significant. Some improvement is achieved over the conventional NNM system by adding the correlation score. The addition of this score is encouraged as it requires minimal additional computation.

7.4.2 Comparison with other Techniques

This section describes the use of HMM models by Forsyth for SV systems to be compared with neural network SV systems. A general description of the DOP technique was given in section 7.3 while the detailed description of it can be found in (Forsyth, 1995). Different performance measures used to assess the DOP technique can be found (Forsyth & Jack, 1993)(Forsyth *et al.*, 1994). The results are presented for text dependent experiments on isolated digits from 25/27 client speakers and 84 impostor speakers. The telephone quality database used is similar for both the HMM and NNM SV systems. The technique of incorporating discriminating observation probabilities (DOP) into an HMM with its various performance measures especially the SI EER will be discussed in this section while the SI ZFR results are discussed in the next section. The parameter sets used for the HMM DOP technique are cepstra, delta cepstra, mel-frequency

cepstral coefficients (MFCC) and delta MFCC. The emphasis of the HMM SV system is the selection of the parameter sets as well the combination of the corresponding DOP scores for further improvement of the SV system. Unless otherwise stated, comparison made in the work carried out in this thesis with HMM models is mainly concerned with the cepstra feature only.

The HMM DOP method has been used to discriminate between a single speaker and a general speaker independent set. According to the results reported, it clearly shows a general increase in performance for all the different features used as measured by EER when DOP are added into the verification system with the exception of delta MFCC features. The best result is obtained with the use of cepstra features followed by MFCC. The corresponding difference parameters show poor performance results. The performance of the proposed NNM methods, SCHMM and the discriminating SCHMM (DOP) SV systems were compared and results are shown in Figure 7.3. Again, the HMM-MLP SV system shows the lowest EER error performance. As observed from this figure the HMM-MLP SV system shows better performance results when compared to the HMM verification system. There is an improvement of 85% when compared to the HMM SV system with no DOP and 75% when DOP is included. The advantages and the disadvantages of the DOP HMM SV systems compared to the NNM SV systems are given in section 7.3. As mentioned before, the HMM will try to optimize the production probability of the time sequence and when used as the input to the classifier this can improve the performance of standard neural networks. In addition to the reported results mentioned in the previous chapter, the data constructed from the output probability of the HMM substantially reduces the number of inputs to the neural network. For example, there are only 2 inputs to the neural network of the system whereas for the NNM with LTN60 there are 120, 122 and 240 inputs to the NNM-C, NNM-CM and NNM-CI respectively. This acts as an advantage to the HMM-MLP SV system thereby reducing the computation required of the neural network during both training and recognition as well as a better generalization capability. It is also noted that 5 hidden nodes are required to

train the HMM-MLP compared to the NNM-CM (8 hidden nodes), NNM-C (8 hidden nodes) and NNM-CI (20 hidden nodes).

The use of the NNM-CM SV system shows an improvement of 17% compared to HMM SV system with DOP and 51% improvement to the HMM SV system with no DOP. This is an indication that the NNM SV method is more robust than the SCHMM SV method based on small amounts of training data. It should be emphasized that the evaluation of the HMM based systems used more client and impostor speakers both in training and testing. The best performance was obtained from the DOP SV system which combines the information from various pair models. It has been reported that the DOP models are superior to the conventional HMM model and that the combination of these models proved useful.

The combination of the two model pair (DOP cepstra plus DOP delta cepstra) (Forsyth, 1995) produced EER of 1.1% in a 12 digit sequence. This is an improvement of 42% over the NNM-CM SV system. The high performance results from the HMM based system is due to the fact that the system is trained with different feature sets. Each feature set has its own speaker discriminating information and some combinations of these features have proven to be optimum. In addition the SV system combined two information sources (conventional HMM score and DOP HMM score) to produce an improved EER.

The objective of this thesis has been to design a SV system using neural networks trained on limited training data. Training a neural network for an SV task using large amounts of labelled speech data is not only time consuming but also requires a huge amounts of training data for better generalization of the neural network. Generalization will depend on the empirical optimization of the neural network structure and the training methods. Thus the design of the SV system using a neural network with this specific objective in mind should follow the criteria

mentioned below to achieve the end results:

- Use preprocessing of the speech signals for a smaller network configuration for ease of implementation and without increase in computational effort.
- Capable of achieving high performance with limited training data.
- Reasonable handling of the temporal information.

Based on the above criteria it can be reasonably concluded that the HMM preprocessor is a better choice than the VQ-based preprocessor to train the MLP classifier for the SV task. The benefit, however must be weighted against the high increased computational requirements using an HMM-preprocessor during training and recognition as well as the increase in computation as more features are added into the processing stage. The results also clarify the effectiveness of the use of the hybrid approach to the SV task. Given the condition with a limited size training database the hybrid discrete HMM-MLP approach outperforms the standard NNM approach as well as the standard SCHMM.

7.4.3 Zero False Rejection (ZFR) Performance Evaluation

Figure 7.4 shows the 12 digit sequence ZFR for each of the techniques used for SV task. The performance of the NNM-CM verification system is also evaluated and compared with the conventional NNM and HMM SV systems. The increase in performance using the NNM-CM by 18.6% and 64.8% (comparison to NNM-C and NNM-CI) respectively suggested that the features learned by the hidden layer during training were adequate to capture the similarity within the speaker group. As established in the previous sections the HMM-MLP SV system shows the best performance results. Comparing this system with the NNM-CI, NNM-C and NNM-CM shows improvement of 72.4%, 36.2% and 21.6% respectively. Although the HMM preprocessor

technique shows significant advantage in system performance, it is interesting to note that such relatively simple NNM systems (excluding NNM-CI) performed nearly as well as the far more complex HMM-MLP SV system.

In the case of the HMM SV system, there is an increase in performance when this system is trained with the inclusion of the DOP as compared to no DOP. When the NNM-CM SV system is compared with the HMM SV system there is 2.3% and 17% performance improvement over the system with DOP and without DOP respectively. When HMM-MLP is compared with HMM SV, there is an improvement of 35% when compared to the HMM SV system with no DOP and 23.3% when DOP is included. For text dependent speaker verification, the vector quantization or HMM preprocessor can be an effective method to deal with speaker dependency in the feature space.

The two performance measures (EER and ZFR) were used to compare different speaker verification techniques which measure different aspects of systems performance. These measures also provide an indication of the possibility of adding similarity scores to the NMM SV system for further enhancement of system performance. They also illustrate the advantage of having the HMM preprocessor for the NNM SV system. The above comparison between different techniques for verification is made possible because of the common database used. As observed by these performance measures, the relative ranking of different systems is based on the relative differences in performance. The ZFR is an additional source of evaluation with emphasis to convenience and the ease of use of the SV system for the client speaker.

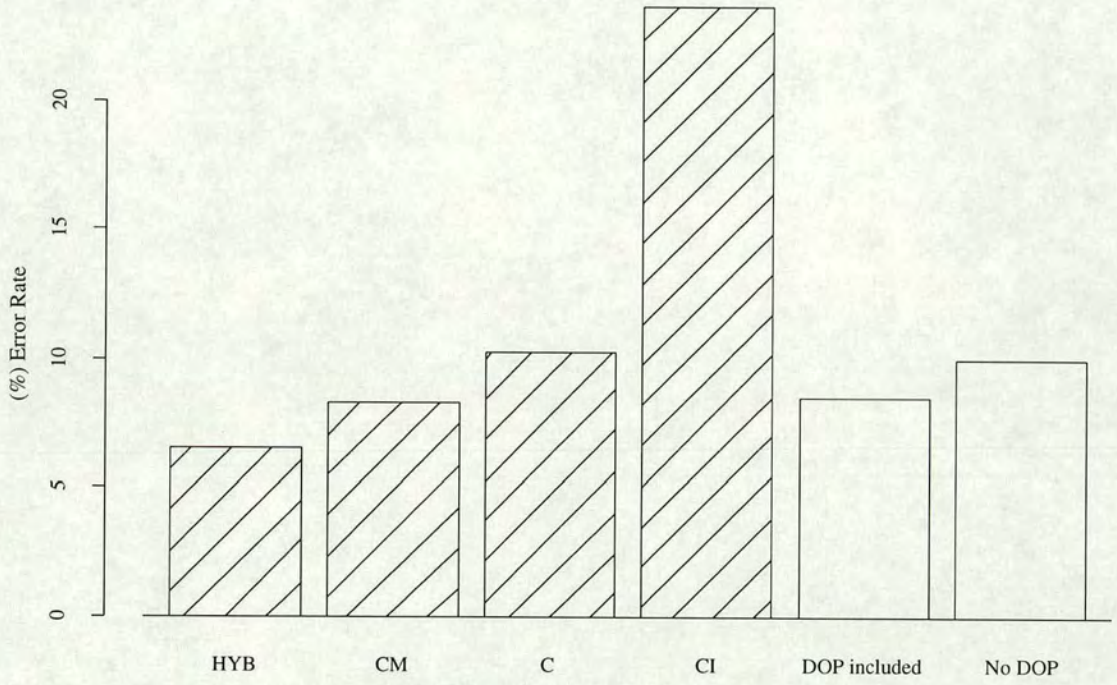


Figure 7.4: Comparison of Percentage Zero False Rejection on 12 Digit Sequences with Speaker Independent Thresholds for Different Techniques Used in SV. a) HYB: HMM-MLP b) CM: NNM-CM c) C: NNM-C d) CI: NNM-CI e) DOP included: SCHMM trained with DOP f) No DOP : SCHMM trained without the DOP

Chapter 8

SUMMARY AND CONCLUSION

8.1 Introduction

The main goal of the research has been to design and select a suitable preprocessor for multi layer neural networks applied to automatic speaker verification with the constraint of limited availability of data. The challenge of the task is that the larger and more complex the input space of a particular pattern becomes, the more training patterns will be required as was outlined in the abstract of this thesis. Chapter 2 has provided the necessary background concerning problems associated with neural networks applied to speaker recognition. Areas for improvement and development based on the reviews in chapter 2 were considered in chapter 3. This research has investigated the use of preprocessors in support to the design of structured networks which converge with limited sample size and practical training times. These techniques have been devised to encourage the application of neural networks to speaker verification.

8.2 Neural Network Speaker Verification Systems

Four different approaches to the design of preprocessors were described for the NNM SV system. Three of them are based on VQ preprocessor and another is the HMM preprocessor. The achievement of the various preprocessors in the study will now be discussed.

8.2.1 Vector Quantization Based Preprocessor

Two methods for creating vector sequences were investigated: these methods make use of a VQ based preprocessor which reduced the amount of input to be fed to the MLP classifier. In NNM-C, the output of the preprocessor for each time frame would contain the index j of the codevector with the minimal distortion and the corresponding distortion value d. In NNM-CI, there are two pairs(j,d) per frame, one for the client codebook and one for the impostor codebook. In most applications for SV using neural networks the inputs to MLP are not preprocessed, creating some difficulties during the training phase. The use of a preprocessor reduced the input vectors and also the network complexity. This can speed the learning process and can facilitate training on limited data. Initially it was assumed that, with the addition of the impostor codebook this would increase the efficiency of the training data as well as its performance. The NNM-CI design works but it is not optimal as the decision surface of this model is more complex due to the complexity of the input to the classifier and also this model has the largest number of connections which may require more training data for better generalization. It may be reasonably assumed that with more training data this model would provide better performance result. However, this is not confirmed due to the nature of the given task. It is reasonable to conclude from the results that the NNM-C model should be used in preference to NNM-CI model when training with limited data. Using the NNM-C also means the amount of input to be fed to MLP is reduced by 50%. Reducing the input vectors to a reasonable size still allows the classification power of neural networks to discriminate between the client and the impostor speakers. This demonstrates the usefulness of the above preprocessing stage in the design of ASV system.

Chapter 5 has addressed one important problem of text dependent NNM-C verification sys-

tem regarding the time normalization of the speech signals. Variations in the temporal word length pose a problem for neural networks with fixed length input layer. Increasing or decreasing the number of inputs to the network will positively affect the performance. Thus, the LTN value chosen as the input to the NNM-C model for this given task is important. Even though time scale variation by LTN may not be the most accurate method to achieve the best match at all points, significant improvement in performance is obtained by appropriately selecting the LTN values for the speaker verification approach used in the system. This means that the simplicity of the approach and ease of implementation gives slight advantage over the other approaches (in comparison to DTW and trace segmentation). It is possible to solve the time scale variation with a more precise procedure (such as dynamic time warping and trace segmentation) which properly matches the internal features of the patterns, however, this may involve excessive computation. The experimental results provided evidence of the possibility of selecting the best LTN values in order to improve the robustness of the NNM-C model.

8.2.2 Hidden Markov Model Based Preprocessor

Further work (chapter 4) was carried out with the use of different preprocessors combined with neural networks to get better results. Much of the earlier effort in speech pattern matching involved aligning the feature vector sequence of an unknown template to the reference template using dynamic time warping. In the HMM based method, the speech pattern is aligned according to the maximum likelihood path or all possible paths through word models. The combination of HMM preprocessor and neural network provided a more optimum solution for the SV task. Specifically, a supervised HMM-Viterbi decoding scheme was trained using speech data and a MLP was trained with the HMM output probabilities. This verification method has the following advantages:

- This approach enables full use of the classification power of neural networks in which maximum likelihood estimation may not be optimal for minimizing classification error rate due to the inaccuracy of the underlying model assumptions.

- It allows the use of multiple information streams to provide maximal use of the information produced by HMM.

- HMM deals with the input vector and statistically models the temporal structure of the speech signals instead of the vector quantization preprocessor. Using a vector quantization preprocessor may not handle properly the time variation of the speech signal since it does not have any statistical modelling capability. The HMM preprocessor then relaxes the severe constraint of fixed input size of the NN classifier.

As a result the HMM preprocessor presents a better solution for use with NNs. The proposed preprocessing method based on HMM improves modelling of output probability estimation and with added information streams helps to enhance the performance of the ASV system. The goal of this thesis was to enhance the performance of SV system by using different methods of preprocessing of the speech signals combined with NNs. The HMM-MLP SV system was found to be superior to the discrete HMM as well as the NNM SV systems trained with the limited data available. However, a sacrifice has to be made in terms of complexity of system and its performance. The complexity of having an HMM preprocessor for the SV system is far higher than the vector quantizer based preprocessor.

8.2.3 Vector Quantization Preprocessor with Similarity Match

The focus of Chapter 6 was to further enhance the performance of the NNM SV system by introducing a client barcode which represents each client in the system. The design of this barcode resulted in a similarity score to be added into the SV system. This system was explored directly

using the experimental procedures established in the previous chapter. A neural network based on the cross match technique was designed and compared to the conventional neural network with a difference: the data set used contained additional information regarding the similarity score and the offset value as opposed to only the data of index (j) and the distortion value (d).

In order to obtain high performance for the verification system the neural network classifier must be able to evaluate the local similarities within an analyzing frame as well as the overall temporal information of the speech data. In situations where a neural network has to classify between the client and the multiple speakers in the group, the error surface can be complex because of the rich spectral variation of the speech samples. The performance of the NNM SV system over time is likely to be affected by these factors. Although the NNM-C system has proven to be successful on the ASV task, the NNM system design is extremely flexible and there are ways where further development can be rewarding. The idea here is the use of correlation values to measure the degree of similarity or dissimilarity between the client and the impostor as added information to aid the learning process. This measure provides a scalar evaluation of how well the client or the impostor utterance correlates with the client barcode. Other information that can be gathered as a result of the development of the client barcode is the offset value. This offset value shows the relative position at which the maximum value of r occurs between the unknown utterance and the client barcode. The new approach has been proven to be successful, but not the approach that used the offset information. An alternative approach had been suggested in the earlier section of chapter 6 to accurately determine the relative position between the unknown utterance and the client barcode that may improve the offset position and thus provide improvement. In this approach also, all samples are considered at the same time (the approach used earlier did not consider all samples but only two samples at a time) to form a single sequence representing the group.

The experiments carried out in chapter 5 applied the NNM-C SV system. The second set of experiments carried out in chapter 6 with the NNM-CM SV system took advantage of the client barcode in order to extract the vital information regarding the speaker. Both experiments have shown that neural network SV systems can be trained to minimize the mean square error between the target output and the desired output vector for a given mapping. The input vectors are either a simple representation or a more complex representation which addresses the similarities within the existing frame as well as the time sequence relationship with respect to the relative position from the main barcode. A second experiment was designed to test the specific hypothesis that the more complex representation serves as a better input for the classification problem. It also tested the ability of the neural network to generalize across speakers.

Integration of the cross match technique within the SV system as a new approach to the preprocessor design provides some advantage in speaker verification performance. Experiments have shown that the system performance can be improved by combining the vector quantized speech data and the similarity measure as the input features. This additional information makes its own weighted contribution to the network. This also shows that the NNM SV system can be further improved by the application of appropriate statistical techniques. The cost of this improvement, however is moderate computation and with no additional training data. The NNM-CM SV results obtained so far are an incentive to go further in the direction presented here. Further suggestions will be made on how to improve the design of the client barcode later in the next section. The strategy employed in chapter 6 provided some assistance to the design of the preprocessor, playing a small but significant part of the whole process of the SV system.

The new preprocessor has been successfully combined with the MLP to build an ASV sys-

tem. The system provides better performance than the NNM-C SV system. Application of the new preprocessor technique gave results comparable with the limited data published in the literature with the HMM SV system. When the NNM-CI is compared to the new technique it shows inferior performance. This does not mean that it is a poor system. Since the amount of speech data cannot be too large, it may be possible that there is not enough samples to train this NNM-CI SV system effectively. The costs of improvement to this model, however, are increased computation and ideally additional training data. This model does not meet the requirement constrained for the design of the ASV system. The possible solution, as adopted in this thesis is the alternative technique based on the NNM-C and NNM-CM for this particular SV application.

8.2.4 Concluding Remarks

The choice of a preprocessor can have a substantial effect on the performance of the system. This is important to the design of a speaker verification system which has to be trained under the restriction of data available. This influences the network size leading to the acceptability of the system. In addition, the choice of the parameter values to train the network as well as the variation in both the client voice and the telephone channels over different recording sessions also contributed to the overall performance of the system. Chapter 4, 5, 6 and 7 have demonstrated various verification systems of different sizes and complexity and have shown the trade off between system size, complexity and performance. In all cases, these approaches are very promising and useful for the SV task. For further studies and modelling of the SV system, they should be evaluated with more client speakers. If the performance of the NNM-CM system is reflective of the actual user population then this can be an acceptable level of initial performance of the system. Based on the same principle the VQ based preprocessors can be extended to

applications in speech recognition.

Despite the practical constraint involved, significant results have been obtained through the NNM SV systems. First, the number of parameters for each model are kept as low as possible so that the requirement for training data can be greatly reduced. Second, the requirement of the speech data for the verification task is kept minimum for user comfort.

8.3 Suggestions For Future Work

Several extensions and improvements to the techniques presented have already been mentioned in the preceding chapters and are given below.

8.3.1 Genetic Algorithm for Neural Network

In determining the optimum network architectures for speaker verification, there are many problems especially for MLP using the back-propagation. Factors to be decided by designers are the application task, network architecture, learning algorithm and the generalization ability of the trained network. After experimentation a design can be identified that can solve a particular task but designing neural networks is actually hard for human beings. A guided search by a human designer as discussed earlier can be inefficient, misdirected, time consuming and costly. There is very little work on evolution and neural networks but current research as described in chapter 2, section **??** seeks to lay the foundation for this approach. For instance genetic algorithms can quickly locate an approximate solution in the weight space and a neural network will perform the hill climbing. Further work should emphasize the use of GA in building a network architecture for speaker verification which is an automated process rather than the traditional way of experimentation.

The genetic algorithm has also been used to select a subset of relevant features (Punch *et al.*, 1993). Often these features will improve the performance of the neural network classifier as well as reducing the computational requirements. The bit string representation of each locus will indicate the presence or the absence of the feature. Study could be conducted using genetic algorithms as a special search method for selecting the best inputs to neural networks. The performance of this genetic algorithm search depends strongly on proper selection of the fitness function for the given task. As argued previously the poor performance of the NNM-CI is because of the complex input feature vector representation to this model. Genetic algorithms can be used to guide the search for optimal combinations of inputs for the neural networks to reach the criteria of fewer inputs. These inputs could then be used to retrain the NNM-CI model for further enhancement of the system with the same amount of data within the required specification.

8.3.2 The Problem of Handset Variations

Previous work has shown that distortions of the speech signals in telephone quality speech are mainly due to transmission characteristics and handset transducers for speaker recognition. The database used in this thesis incorporates a large sampling of measurable distortions from a variety of handsets available as well as the telephone channel across the telephone network in United Kingdom. The ASV system design is not aimed specifically at coping with clients using a variety of handsets. It is rather difficult to enforce microphone variation explicitly. However, subjects were encouraged to use different types of handsets during the collection of this speech database. This is one area that has not been addressed properly in the thesis. The handset types are not known for the BRENT corpus. An approach similar to (Reynolds, 1997) could be used to address the problem of handset variability. This makes use of the HTIMIT corpus released by

LDC to generate a handset detector. Detailed description of the HTIMIT corpus can be found in chapter 2, section 2.3.3. Initially, cepstral mean subtraction and RASTA filtering can be applied to minimize the linear filter effects. Then, a maximum likelihood classifier based on Gaussian mixture or a neural network classifier can be built to discriminate between speech originating from a carbon handset and speech generated from an electret handset. The models should be trained with speech from the same speakers so that the differences between the models should only be attributed to uncompensated transducer effects. If, however, a neural network model is chosen for this task it has the advantage of being a single model with multiple outputs used for classification of any number of classes. The classes in this case will depend on the different types of handsets used in the experiments. This handset detector can then be applied to the BRENT corpus to label the utterances as either carbon button or electret for further improvement of the speaker verification task. The experimental conditions in this case will deal with handset mismatch between training and testing which is known to give serious performance degradations for speaker verification (Rosenberg *et al.*, 1992)(Rosenberg *et al.*, 1994)(Naik, 1994)(Heck & Weintraub, 1997).

8.3.3 Improvement to the Development of Client Barcode

A new method based on the development of a client barcode with a cross match technique was proposed to combat intra-speaker variations. Chapter 6 provided an extensive discussion of the development of the new method as well as its performance for the given task. Clearly the use of such method in automatic speaker verification is limited, but, by providing additional information regarding aspects of inter and intra speaker variability which aid speaker discrimination, it may well offer a valuable solution for further research. Further improvement to the design of the barcode suggested in section 6.3 could be made by performing modification to its algorithm

according to (Linton & Zainodin, 1987). Modification can be done in Step 2 of the algorithm. Here, instead of comparing only two tokens, all tokens are considered at the same time and the current similarity values are sorted in order. A criteria is set so that samples having similarities greater than some threshold value will form a group. That is if the similarity between two sequences is greater than this threshold the two sequences will be merge, but not otherwise. In Step 3, decrement the threshold and the entries are updated by adding and deleting the tokens and the sequence during the process. In order to merge these data into sequences equation 6.5 must be satisfied. The statistical technique to measure the similarity between two sequences or tokens is the correlation coefficient calculated from equation 6.2. There are other possible ways to measure the similarity values between two speech signals and these are given below:

$$t = r(X_{1j}, X_{2j}, p)\sqrt{\frac{n-2}{1-r(X_{1j}, X_{2j}, p)^2}} \tag{8.1}$$

or

$$z = \frac{1}{2}\sqrt{n-3}\quad \ln\left(\frac{1+r(X_{1j}, X_{2j}, p)}{1-r(X_{1j}, X_{2j}, p)}\right) \tag{8.2}$$

where n is the number of indices of overlaps at this position.

A further consideration in the development of the barcode for the speaker verification system is to introduce a dissimilarity measure. If the mechanisms which go to produce this dissimilarity can be clearly defined, they can then be incorporated into the algorithm based on the given

equation:

$$r_{new} = r_{old} - \text{Dissimilarity} \tag{8.3}$$

where r_{old} is the correlation coefficient calculated in equation 6.2.

This a reasonable consideration as there is the possibility where two similarities scores (each similarity score can come from different sequences) can have the same value when all tokens are considered in the process. Thus, the idea of having dissimilarity introduced to the above algorithm will not allow such case to happen. A dynamic programming technique could be used to obtain the dissimilarity value or the dissimilarity value can be based on the differences between the client barcode and the impostor barcode correlation scores. These suggestions so far have not been attempted in this thesis. The above modified algorithm would be able to generate a more reliable client barcode. Most importantly this approach considers the variable input vector size. From the viewpoint of the above analysis this research represents a further possible dimension for speaker verification system design, an approach which deserves further attention.

Appendix A

Publications

1. S. Hussain, F. R. McInnes and M. A. Jack, "Enhanced Automatic Speaker Verification Based on a Combination of Hidden Markov Models and Multi Layer Perceptrons", 2nd IEEE Malaysia International Conference On Communication, Vol. 2, pp. 831-834, 1995.

2. S. Hussain, F. R. McInnes and M. A. Jack, "Speaker Verification Using Hidden Markov Models and Multi Layer Perceptrons", Postgraduate Journal of The Department of Electrical Engineering, University of Edinburgh, Issue 2, January, 1996.

3. S. Hussain, F. R. McInnes and M. A. Jack, "Comparison Of Neural Network Techniques For Speaker Verification", Proceedings of The Sixth Australian International Conference on Speech Science and Technology (SST-96), pp. 245-250, 1996.

4. S. Hussain, F. R. McInnes and M. A. Jack, "Improved Speaker Verification System With Limited Training Data On Telephone Quality Speech", Proceedings of The 5th European Conference on Speech Communication and Technology, Vol 2, pp. 835-838, September, 1997.

Appendix B

Speech Databases for Speaker Verification

SpeechDat

The Commission of the European Communities funded the SpeechDat (Hoge *et al.*, 1997) project aimed at having databases for European Union languages as well as some major dialectal variants and minority languages. One important contribution of this project is that it provide a realistic environment for speaker recognition. The SpeechDat database is divided into three different sets. The first two databases are related to speech data collected over fixed telephone network and mobile telephone network. The final database is concerned with speaker verification. The speaker verification database is further divided into three sets with each database consisting of 20 to 120 speakers in each language. There are 20 to 50 calls per speaker. The SpeechDat databases complement each other in such a way that the first two databases represent speaker verification impostor testing material while the speaker verification database is a collection of speaker dependent material that can be used to train and test the SV model. The speech materials gathered include digit and letter sequences, numbers and money amounts, common application key words and wordspotting phrases, dates, times, yes/no responses, person and city directory assistance names and phonetically rich words and sentences.

YOHO

A comprehensive SV database is available in the YOHO corpus. It consisted of "combination lock" number sequences (for example 36-24-36) in 4 enrolment sessions and 10 test sessions per subject with 4 phrases per session (James *et al.*, 1997). The disadvantage of this database is that it was not recorded over the telephone channel and thus it does not provide the realistic environment for telephone quality speech needed for SV system design.

KING

Another database which can be used is the KING database, principally designed for experiments

in text independent speaker identification or verification over "toll quality" telephone line (Godfrey *et al.*, 1994). There are two versions of this database where there are 51 male speakers which differ in channel characteristics: one in terms of telephone handset and the other version from a high quality microphone. The intervals between collection of these data varied from a week to months.

HTIMIT

Originating from the well known TIMIT database specifically designed for speech recognition, a transformation is done from the original database to suit the application of a telephone quality speech database. This large database is designed to highlight the effects of handset transducer. Discussions on the use of such database can be found in section 2.3.3.

Other collections of database which are worth mentioning include SIVA, an Italian database (Falcone & Gallo, 1996), GANDALF a Swedish Telephone database (Melin, 1996) and BRENT, from British Telecom (Forsyth, 1995). The application of the BRENT speaker verification database is further discussed in chapter 3, section 3.3. The use of this database in the design of the NN SV system should be able to cope with clients using different types of telephones and thus creating an environment with a variety of microphone types.

Being aware of the impressive evolution of speech recognition technology, the speaker recognition community have taken similiar steps to introduce a standardized database for speaker recognition performance evaluation. Even though there is a huge range of speech recognition databases, it is still necessary to have speaker recognition databases mainly to solve the intra-speaker variability issue. These databases normally include multiple recordings from each speaker spread over a specific period in order to capture long term variations of the speech signal. A brief description of the available SV databases has been introduced above. The number of standardized speech databases have significantly contributed to the steady progress of speaker recognition technology. It is hoped that this will further develop the area of speaker recognition especially in speaker verification in various parts of the world. This progress will eventually lead to a number of practical and productive applications of SV closely related to the telephone industry.

The nature of speech material used will depend on the application intended. One major consideration as to the types of text on which the SV system must operate can be prescribed by the client. Some customers prefer the use of digit strings rather than names or specific phrases. Customers also prefer to choose their own password. One advantage of using digit strings is the capability of combining recognition and verification of voice password in a single task (Naik, 1994). The choice of using fixed text has certain advantages such as to help reduce the amount of processing and also reduce intra-speaker variability. The disadvantage with a short fixed text is that it can be defeated by pre-recording of an authorised client and random prompt utterances or the text prompted systems are suggested as an alternative. In a free text system the choice of text is less constrained and normally has longer utterances. The system is designed to cope with a variety of speech styles and utterance durations. It is more difficult to reliably test such systems and they are less desirable for telephonic applications (Markel & Davis, 1979)(Hunt, 1983).

The actual contents used in the speaker verification experiments are largely dependent on the application intended. Thus digit names "one", "two" to "nine" and familiar names are commonly used as voice passwords. Command words such as "yes/no", "add", "enter", "change" and "erase" are popular for computer text processing. In mobile communication familiar names, isolated digits, voice passwords and command words (directory, redial, call, etc) are the types of texts normally incorporated into the SV system. A review by (Rooney, 1990) indicates there are studies which select words beginning with a plosive and ending with a non plosive to facilitate word boundary detection and sentences and phrases that contain only phonemically voiced segments. In another example, all oral sentence "We were away a year ago" provided better performance compared to text containing nasal segments such as "I know when my lawyer is due".

References

Anderson, T., & Patterson, R. (1994). Speaker Recognition With The Auditory Image Model and Self Organization Feature Maps: A Comparison With Traditional Techniques. *Pages 153–156 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Artieres, T., & Gallinari, P. (1995 September). Multi-State Predictive Neural Networks For Text Independent Speaker Recognition. *Pages 633–636 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Atal, B. (1972). Automatic Speaker Recognition Based On Pitch Contours. *Pages 1687–1697 of: The Journal of the Acoustic Society of America*, vol. 52.

Atal, B. (1974). Effectiveness of Linear Prediction Characteristics of The Speech Wave For Automatic Speaker Identification and Verification. *Pages 1304–1313 of: The Journal of the Acoustic Society of America*, vol. 55.

Atal, B.S. (1976 April). Automatic Recognition Of Speakers From Their Voices. *Pages 460–474 of: Proceedings of The IEEE*, vol. 64.

Barger, P., Slomka, S., Castellano, P., & Sridharan, S. (1996 December). Gender Gates For Automatic Speaker Recognition. *Pages 19–24 of: Proceedings of the Australian International Conference on Speech Science and Technology*, vol. 1.

Beaufays, F., & Weintraub, M. (1997 March). Model Transformation For Robust Speaker Recognition From Telephone Data. *Pages 1063–1066 of: Voice+ The European Magazine For Applications Of Computer Telephony*, vol. 2.

Bengio, Y., Mori, R., & Flammia, G. (1992 March). Global Optimization of a Neural Network-Hidden Markov Model Hybrid. *In: IEEE Trans on Neural Network*, vol. 3.

Bennani, Y. (1993 March). Probabilistic Cooperation Of Connectionist Expert Modules: Validation On A Speaker Identification Task. *Pages 541–544 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Bennani, Y., & Gallinari, P. (1991 May). On The Use Of TDNN-Extracted Features Information In Talker Identification. *Pages 393–396 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Bennani, Y., & Gallinari, P. (1994 April). Connectionist Approach for Automatic Speaker Recognition. *Pages 95–102 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Bennani, Y., Soulie, F., & Gallinari, P. (1990 April). A Connectionist Approach For Automatic Speaker Identification. *Pages 265–268 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Bernasconi, C. (1990). On Instantaneous And Transitional Spectral Information For Text Dependent Speaker Verification. *Pages 129–139 of: Speech Communication*, vol. 9.

Bonifas, J., Rioja, I., Gonzalez, B., & Saoudi, S. (1995 September). Text Dependent Speaker Verification Using Dynamic Time Warping And Vector Quantization Of LSF. *Pages 359–362 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Boulard, H., & Wellekens, C. (1989). Speech Pattern Discrimination And Multilayer Perceptrons. *Pages 1–19 of: Computer Speech and Language*, vol. 3.

Bourlard, H. (1991 September). Neural Nets And Hidden Markov Models: Review And Generalization. *Pages 363–369 of: Proceedings of the European Conference on Speech Technology*, vol. 2.

Bourlard, H., Boite, J., D'hoore, B., & Saerens, M. (1993 September). Performance Comparison of Hidden Markov Models and Neural Networks for Task Dependent and Independent Isolated Word Recognition. *Pages 1925–1928 of: Proceedings of the European Conference on Speech Technology*, vol. 3.

Bourlard, H., Hermansky, H., & Morgan, N. (1996). Towards Increasing Speech Recognition Error Rates. *Pages 205–231 of: Speech Communication*, vol. 18.

Buck, J., Burton, D., & Shore, J. (1985 March). Text-Independent Speaker Recognition Using Vector Quantization. *Pages 391–394 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Burr, D.J. (1988). Speech Recognition Experiments With Perceptron. *Pages 144–153 of:* Anderson, D. (ed), *Neural Information Processing System.* American Institute of Physics, New York.

Calonge, T., Alonso, L., Ralha, R., & Sanchez, L. (1995 September). Parallel Implementation of an Hybrid Neural Networks Used For Speech Recognition Task. *Pages 153–156 of: Proceedings of the European Conference on Speech Technology.*

Carey, M., Parris, E., & Bridle, S. (1991 May). A Speaker Verification Using Alpha-Nets. *Pages 397–400 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Carey, M., Parris, E., Thomas, H., & Bennett, S. (1996). Robust Prosodic Features For Speaker Identification. *Pages 1800–1803 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 3.

Cerf, P.L., & Compernolle, D.V. (1993 April). Using Parallel MLPs As Labelers For Multiple Codebook HMMS. *Pages 561–564 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Che, C., & Lin, Q. (1995 September). Speaker Recognition Using HMM With Experiments On the YOHO Database. *Pages 625–628 of: Proceedings of the European Conference on Speech Technology.*

Ching, P., Chow, K., Lee, T., Ng, A., & Chan, L. (1997 April). Development Of A Large Vocabulary Speech Database For Cantonese. *Pages 1775–1778 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Chou, W., Juang, H., & Lee, C.H. (1993 April). Minimum Error Rate Training Based on the N-Best String Models. *Pages 652–655 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Ciria, R., de Sotomayor, R., Aguila, C., Parera, J., & Santos, J. (1995 September). Voice Processing Architecture For Computer Telephony Integration. *Pages 63–66 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Clarke, G.M., & Cooke, D. (1994). *A Basic Course in Statistics*. Edward Arnold.

Cox, S.J. (1988 April). Hidden Marhov Models for Automatic Speech Recognition: Theory and Application. *In: British Telecom Technical Journal*, vol. 6.

Das, S., & Mohn, W. (1971). A Scheme For Speech Processing In Automatic Speaker Verification. *Pages 32–43 of: IEEE Trans on Audio and Electroacoustics*, vol. 1.

de Veth, J., & Bourlard, H. (1994 April). Comparison of Hidden Markov Model Techniques For Automatic Speaker Verification. *Pages 11–14 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

de Veth, J., Gallopyn, G., & Bourlard, H. (1993 March). Limited Parameter Hidden Markov Models For Connected Digit Speaker Verification Over Telephone Channels. *Pages 247–250 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Demichelis, P., Fissore, L., Laface, P., Micca, G., & Piccolo, E. (1989 May). On The Use of Neural Network for Speaker-Independent Isolated Word Recognition. *Pages 314–317 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Doddington, G. R. (1985). Speaker Recognition : Identifying People By Their Voices. *Pages 1651–1664 of: Proceedings of The IEEE*, vol. 73.

Eberhart, R., & Dobbins, R. (1990). *Neural Networks PC Tools*. Academic Press.

Epraim, Y., & Rabiner, L. (1988). On The Relations Between Modelling Approachs For Information Sources. *Pages 24–27 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Falcone, M., & Gallo, A. (1996 October). The "SIVA" Speech Database For Speaker Verification: Description And Evaluation. *Pages 1902–1905 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 3.

Farrell, K., & Mammone, R. (1994 April). An Evaluation Of Supervised And Unsupervised Classifiers For Speaker Recognition. *Pages 67–70 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Federico, A., Ibba, G., & Paoloni, A. (1987). A New Automated Methods For Reliable Speaker Identification And Verification Over Telephone Channels. *Pages 1457–1460 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 3.

Forsyth, M. (1995). *Semi-Continuous Hidden Markov Models For Automatic Speaker Verification*. Ph.D. thesis, University of Edinburgh.

Forsyth, M., & Jack, M. (1993). Discriminating Semi Continuous HMM For Speaker Verification. *In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Forsyth, M., Sutherland, A., Elliot, J., & Jack, M. (1993). HMM Speaker Verification With Sparse Training Data On Telephone Quality Speech. *Pages 411–416 of: Speech Communication*, vol. 13.

Forsyth, M., Bagshaw, P. C., , & Jack, M. A. (1994). Incorporating Discriminating Observation Probabilities (DOP) Into Semi-Continuous HMM For Speaker Verification. *Pages 19–22 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Fredrickson, S., & Tarassenko, L. (1994 April). Radial Basis Functions For Speaker Identification. *Pages 107–110 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Furui, S. (1981a). Cepstral Analysis Technique For Automatic Speaker Verification. *Pages 254–272 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. COM 29 no2.

Furui, S. (1981b). Comparison Of Speaker Recognition Methods Using Statistical Features And Dynamic Features. *Pages 342–350 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP 29 no3.

Furui, Sadaoki. (1989). *Digital Speech Processing, Synthesis and Recognition.* Marcel Dekker Inc, New York.

Furui, Sadaoki. (1991 December). Speaker Dependent Feature Extraction, Recognition And Processing Technique. *Pages 505–520 of: Speech Communication.*

Furui, Sadaoki. (1994 April). An Overview Of Speaker Recognition Technology. *Pages 1–9 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Furui, Sadaoki. (1997 March). Recent Advances In Speaker Recognition. *Pages 237–252 of: First International Conference, AVBPA'97, Crans-Montana, Switzerland.*

Gaganelis, D., & Frangoulis, E. (1991 May). A Noval Approach To Speaker Verification. *Pages 373–376 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Gauvain, J., Mariani, J., & Lienard, J. (83 April). On The Use Of Time Compression For Word-Based Recognition. *Pages 1029–1032 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

George, M. (1995 October). A New Approach To Speaker Verification. *Pages 51–60 of: Voice+ The European Magazine For Applications Of Computer Telephony*, vol. 2.

Gish, H., Krasner, M., Karnofsky, K., S.Roucos, Schwartz, R., & Wolf, J. (1985 March). Investigation Of Text-Independent Speaker Identification Over Telephone Channels. *Pages 379–382 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Gish, H., Krasner, M., Russell, W., & Wolf, J. (1986 April). Methods And Experiments For Text-Independent Speaker Recognition Over Telephone channels. *Pages 865–868 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Godfrey, J., Graff, D., & Martin, A. (1994 April). Public Databases for Speaker Recognition And Verification. *Pages 39–42 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Gong, Yifan. (1995 September). Evaluation of Bayes Decision Approach To Automatic Determination Of Thresholds For Speaker Verification. *Pages 367–370 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Gray, R. M. (1984 April). Vector Quantization. *Pages 4–28 of: IEEE ASSP Magazine.*

Hammerstorm, D. (1993 July). Working With Neural Networks. *In: IEEE Spectrum.*

Hangai, S., Shigetoshi, S., & Miyauchi, K. (1990). Speaker Identification Based On Multipulses Glottal And LPC vocal -Tract Model. *Pages 1269–1272 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 2.

Hangai, S., Shigetoshi, S., & Miyauchi, K. (1992 October). Speaker Verification Using Locations And Sizes of Multipulses on Neural Network. *Pages 1439–1442 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 2.

Hattori, H. (1992 March). Text Independent Speaker Recognition Using Neural Networks. *Pages 153–156 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Hattori, Hiroaki. (1994 April). Text-Independent Speaker Verification Using Neural Networks. *Pages 103–106 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Hayakawa, S., & Itakura, F. (1994 April). Text Dependent Speaker Recognition Using The Information In The Higher Frequency Band. *Pages 137–140 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Hayakawa, S, Takeda, K., & Itakura, F. (1997 March). Recent Advances in Speaker Identification Using Harmonic Structure of LP-Residual Spectrum. *Pages 253–260 of: First International Conference, AVBPA'97, Crans-Montana, Switzerland.*

Haykin, Simon. (1994). *Neural Networks - A Comprehensive Foundation.* Macmillan College Publishing Company.

He, J., Liu, L., & Palm, G. (1994 September). A Text-Independent Speaker Identification System Based On Neural Networks. *Pages 1851–1854 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 4.

He, J., Liu, L., & Palm, G. (1995 September). On The Use Of Features From Prediction Residual Signals In Speaker Identification. *Pages 313–316 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Hecht, Nielsen Robert. (1991). *Neurocomputing.* Addison - Wesley Publishing Company.

Heck, L., & Weintraub, M. (1997 April). Handset Dependent Background Models For Robust Text-Independent Speaker Recognition. *Pages 1071–1074 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Hermansky, H., Morgan, N., Bayya, A., & Kohn, P. (1991 September). Compensation For The Effect of The Communication Channel In Auditory-Like Analysis of Speech (RASTA-PLP). *Pages 1367–1370 of: Proceedings of the European Conference on Speech Technology*, vol. 3.

Higgins, A., Bahler, L., & Porter, J. (1993 April). Voice Identification Using Nearest-Neighbour Distance Measure. *Pages 375–378 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Hoge, H., Tropf, H., Winski, R., Heuvel, H., Umbach, R., & Choukri, K. (1997 April). European Speech Databases For Telephone Applications. *Pages 1771–1774 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 3.

Holmes, J. N. (1995). *Speech Synthesis And Recognition.* 2-6 Boundary Row, London SE1 8HN, UK: Chapman And Hall.

Huang, W., Lippmann, R., & Gold, B. (1988). A Neural Net Approach To Speech Recognition. *In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Huang, X., & Jack, M. (1988). On Several Problems Of Hidden Markov Models. *Pages 17–22 of: Proceedings Speech 88, 7th FASE Symposium, Edinburgh.*

Huang, X., Ariki, Y., & Jack, M. (1990). *Hidden Markov Models For Speech Recognition.* 22 George Square, Edinburgh: Edinburgh University Press.

Hunt, A. (1991 September). New Commercial Applications of Telephone-Network-Based Speech Recognition and Speaker Verification. *Pages 431–434 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Hunt, M. (1983 March). Investigation of Text-Independent Speaker Recognition Over Communications Channels. *Pages 563–566 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Hush, D., & Horne, B. (1993 January). Progress In Supervised Neural Networks. *In: IEEE Signal Processing Magazine.*

Imperl, B., Kacic, Z., & Horvat, B. (1997 April). The Use of Harmonic Features In Speaker Recognition. *Pages 1131–1134 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Ishak, I., Hussain, S., & Zulkarnian, M. (1992). PC-Based Speaker Dependant Isolated Malay Speech Recognition System. *In: Journal Elektrika.* 2/92, vol. 5. Universiti Teknologi Malaysia.

James, D., Hutter, H., & Bimbot, F. (1997 March). The CAVE Speaker Verification Project-Experiments on the YOHO and SESP Corpora. *Pages 385–394 of: First International Conference, AVBPA'97, Crans-Montana, Switzerland.*

Jou, I., Lee, Su., Lin, Min., Tseng, Chih., Yu, Shih., & Tsay, Yuh. Juaain. (1990 November). A Neural Network Based Speaker Verification System. *Pages 1273–1276 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 2.

Kao, Y., Baras, J., & Rajasekaran, P. (1993 March). Robustness Study of Free Text Speaker Identification And Verification. *Pages 379–382 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Kilmartin, L., & Ambikairajah, E. (1992 December). A Hybrid MLP-RBF Based Speaker Verification System. *Pages 73–78 of: Proceedings of the Australian International Conference on Speech Science and Technology*, vol. 1.

Lastrucci, L., Gori, M., & Soda, G. (1994 April). Neural Autoassociators For Phoneme-Based Speaker Verification. *Pages 189–192 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Lee, C., & Gauvain, J. (1993 April). Speaker Adaptation Based On MAP Estimation Of Hidden Markov Model Parameters. *Pages 558–561 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Lee, K.F. (1988). *Large-Vocabulary Speaker Independent Continuous Speech Recognition: The SPHINX System.* Ph.D. thesis, Computer Science Department, Carnegie Mellon University.

Li, H., Haton, J., & Gong, Y. (1995 September). On MMI Learning of Gaussian Mixture For Speaker Models. *Pages 363–366 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Linde, Y., Buzo, A., & Gray, R. M. (1980). An Algorithm For Vector Quantizer Design. *In: IEEE Trans on Communication*, vol. 3.

Linton, C. D., & Zainodin, H. J. (1987). Grouping Techniques in Dendrochonology. *Pages 14–25 of: Science and Archaeology*, vol. 29.

Lippmann, R., & Gold, B. (1987). Neural Classifiers Useful for Speech Recognition. *In: Proceedings of the First International Conference on Neural Networks.*

Lippmann, R., & Singer, E. (1993 April). Hybrid Neural Networks/HMM Approaches To Wordspotting. *Pages 565–568 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Lippmann, R.P. (1987 April). An Introduction To Computing With Neural Networks. *In: IEEE ASSP Magazine.*

Lippmann, R.P. (1988). Neural Networks For Computing. *In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Lippmann, R.P. (1989). Review Of Neural Networks For Speech Recognition. *Pages 1–38 of: Neural Computation*, vol. 1.

Liu, C., Lin, M., Wang, W., & Wang, H. (1990 April). Study Of Line Spectrum Pair Frequencies For Speaker Recognition. *Pages 277–280 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Liu, C., Lee, C., Juang, B., & Rosenberg, A. (1994 April). Speaker Recognition Based On Minimum Error Discriminative Training. *Pages 325–328 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Logan, E., Wrench, A., Sutherland, A., & Jack, M. (1990). *A Real Time Speaker Verification System Using Hidden Markov Model.* Tech. rept. University of Edinburgh. Centre for Speech Technology Research.

Luck, J. E. (1969). Automatic Speaker Verification Using Cepstral Measurement. *Pages 1026–1032 of: The Journal of the Acoustic Society of America*, vol. 46.

Makhoul, J. (1974). Linear Prediction In Automatic Speech Recognition. *In: IEEE Symposium On Speech Recognition.* Academic Press.

Maren, A.J, Harston, Craig, & Pap, Robert M. (1990). *Handbook of Neural Computing Application.* Academic Press.

Markel, J. (1974). An Optimal Linear Prediction Synthesizer Structure For Array Processor Implementation. *In: IEEE Symposium On Speech Recognition.* Academic Press.

Markel, J., & Davis, S. (1979). Text-Independent Speaker Recognition From A Large Linguistically Unconstrained Time-Spaced Data Base. *Pages 74–82 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP-27 no1.

Markel, J., Oshika, B., & Gray, A. (1977). Long-Term Averaging For Speaker Recognition. *Pages 330–337 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP-25 no4.

Markowitz, J. (1997 March). Voice Identification Finds Hope In Security. *Pages 23–25 of: Voice+ The European Magazine For Applications Of Computer Telephony*, vol. 2.

Matsui, T., & Furui, S. (1992 October). Speaker Recognition Using Concatenated Phoneme Models. *Pages 603–606 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 1.

Matsui, T., & Furui, S. (1993 April). Concatenated Phoneme Models For Text Variable Speaker Recognition. *Pages 391–394 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Matsui, T., Kanno, T., & Furui, S. (1995 September). Speaker Recognition Using HMM Composition In Noisy Environments. *Pages 621–624 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

McInnes, F. R. (1988). *Adaptation Of Reference Patterns In Word-Based Speech Recognition.* Ph.D. thesis, University of Edinburgh.

Melin, H. (1996 October). GANDALF - A Swedish Telephone Speaker Verification Database. *Pages 1954–1057 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 3.

Morgan, D. P., & Scofield, C. L. (1993). *Neural Networks And Speech Processing.* Kluwer Academic Publishers.

Myers, C., Rabiner, L., & Rosenberg, A. (1980). Performance Tradeoffs In Dynamic Time Warping Algorithms For Isolated Word Recognition. *Pages 623–635 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. 28.

Naik, J. (1994 April). Speaker Verification Over The Telephone Network: Databases, Algorithms And Performance Assessment. *Pages 31–38 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Naik, J., Lorin, L., & Doddington, G. (1989 May). Speaker Verification Over Long Distance Telephone Lines. *Pages 524–527 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Naik, Jay, & Lubensky, D. (1994 April). A Hybrid HMM-MLP Speaker Verification Algorithm For Telephone Speech. *Pages 153–156 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Nakagawa, S., & Markov, K. (1997 March). Speaker Verification Using Frame And Utterance Level Likelihood Normalization. *Pages 1087–1090 of: Voice+ The European Magazine For Applications Of Computer Telephony*, vol. 2.

Nakamura, S., & Akabane, T. (1991 May). A Neural Speaker Model For Speaker Clustering. *Pages 853–856 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Ng, K., Su, J., & Xu, B. (1995 September). Speaker Recognition With Discriminative Speaker VQ Models. *Pages 325–328 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Normandin, Y., Cardin, R., & Mori, R. (1994). High Performance Connected Digit Recognition Using Maximum Mutual Information Estimation. *Pages 299–311 of: IEEE Trans. on Acoustic, Speech, and Signal Processing.*

Oglesby, J., & Mason, J. S. (1990 April). Optimization Of Neural Models For Speaker Identification. *Pages 261–264 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Oglesby, J., & Mason, J. S. (1991 May). Radial Basis Function For Speaker Recognition. *Pages 393–396 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Oglesby, J., & Mason, S. (1988 August). Speaker Identification Using Neural Nets. *Pages 1357–1363 of: Proceedings Speech 88, 7th FASE Symposium, Edinburgh*, vol. 4.

Olive, J. (1971). Automatic Formant Tracking By A Newton Raphson Technique. *Pages 661–670 of: The Journal of the Acoustic Society of America*, vol. 50.

Openshaw, J., & Mason, J. (1994). Optimal Noise-masking of Cepstral Features For Robust Speaker Identification. *Pages 231–234 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Oscal, T. C., Sheu, B.J., & Fang, Wai-Chi. (1992 March). Adaptive Vector For Image Compression Using Self-Organization Approach. *Pages 385–388 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Owens, F. J. (1993). *Signal Processing Of Speech.* Houndmills, Basingstoke, Hampshire RG21 2XS and London: The Macmillan Press Ltd.

Pager, R., Harrison, T., & Fallside, F. (1986). Boltzman Machine For Speech Recognition. *Pages 2–27 of: Computer Speech and Language*, vol. 1.

Pan, K., Soong, F., & Rabiner, L. (1985). A Vector Quantization - Based Preprocessor For Speaker Independent Isolated Word Recognition. *Pages 169–175 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. COM 28 no1.

Paoloni, A., Ragazzini, S., & Ravaioli, G. (1997 March). Text-Independent Speaker Verification Using Multiple State Predictive Neural Networks. *Pages 279–286 of: First International Conference, AVBPA'97, Crans-Montana, Switzerland.*

Parson, T. (1986). *Voice And Speech Processing.* New York: Mc Graw Hill Inc.

Patel, J. (1997 April). Low Complexity VQ For Multi-Tap Pitch Predictor Coding. *Pages 763–766 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Peeling, S., & Moore, R. (1988a). Experiments In Minimally Distinct Word-Pair Discrimination Using The Multi-layer Perceptron Using Multilayer Perceptron. *In: EURASIP Journal Speech Communication*, vol. Royal Signals and Radar Establishment.

Peeling, S., & Moore, R. (1988b). Isolated Digit Recognition Using The Multilayer Perceptron. *In: EURASIP Journal Speech Communication*, vol. 7.

Peeling, S., Moore, R., & Tomlinson, M. (1986). The Multilayer Perceptron As A Tool for Speech Pattern Processing Research. *Pages 307–314 of: Proceedings of the International Conference on Speech and Hearing*, vol. 8.

Peeling, S., Moore, R., & Varga, A.P. (1987). *Isolated Digit Recognition Using Multilayer Perceptron.* Proceeding NATO ASI Speech Understanding.

Perdue, R., & Scherer, J. (1996 September). The Way We Were: Speech Technology, Platforms and Applications In The "old" ATT. *In: IEEE Workshop Interactive Voice Technology For Telecommunications Applications*, vol. 1.

Picone, J. (1990 July). Continuous Speech Recognition Using Hidden Markov Model. *In: IEEE ASSP Magazine.*

Poritz, A. (1982). Linear Predictive Hidden Markov Models And The Speech Signal. *Pages 1291–1294 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Punch, W., Goodman, E., Pei, M., Shun, L., Hovland, P., & Enbody, R. (1993). Further Research On Feature Selection And Classification Using Genetic Algorithms. *Pages 557–564 of: Proceedings of the International Conference on Genetic Algorithm.* Champaign ILL.

Rabiner, L., & Juang, B-H. (1993). *Fundamentals Of Speech Recognition.* Englewood Cliffs, New Jersey 07632: Prentice Hall International Editions.

Rabiner, L., Levinson, S., & Sondhi, M. (1983 April). On The Application Of Vector Quantization And Hidden Markov Model To Speaker Independent Isolated Word Recognition. *In: The Bell System Technical Journal*, vol. 62.

Rabiner, L.R. (1986 January). An Introduction To Hidden Markov Model. *In: IEEE ASSP Magazine*, vol. 3.

Renals, S., & Morgan, N. (1992 December). *Connectionist Probability Estimator In HMM Speech Recognition.* Tech. rept. International Computer Science Institute, 1947 Center Street, Suite 600,Berkeley,California 94704.

Reynolds, D. (1997 April). HTIMIT and LLHDB: Speech Corpora For The Study Of Handset Transducer Effects. *Pages 1535–1538 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Reynolds, D., & Rose, R. (1992 March). An Integrated Speech Background Model For Robust Speaker Identification. *Pages 185–188 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Reynolds, D. A., & Carlson, B.A. (1995 September). Text-Dependent Speaker Verification Using Decoupled and Integrated Speaker and Speech Recognizer. *Pages 647–650 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Rigoll, G. (1989). Speaker Adaptation For Large Vocabulary Speech Recognition System Using "Speaker Markov Models". *Pages 5–8 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Rigoll, G. (1990). Baseform Adaptation for Large Vocabulary Hidden Markov Model Based Speech Recognition Systems. *Pages 141–144 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rigoll, G. (1992 March). Unsupervised Information Theory-Based Training Algorithms for Multilayer Neural Networks. *Pages 393–396 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rooney, E.D. (1990). *Nasality in Automatic Speaker Verification.* Ph.D. thesis, University of Edinburgh.

Rose, R., & Reynolds, D. (1990 April). Text Independent Speaker Identification Using Automatic Acoustic Segments. *Pages 293–296 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rosenberg, A., & Sambur, M. (1975). New Techniques For Automatic Speaker Verification. *In: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP-23 no2.

Rosenberg, A., & Soong, F. (1986 April). Evaluation Of A Vector Quantization Talker Recognition System In Text Independent And Text Dependent Modes. *Pages 873–876 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Rosenberg, A., & Soong, F. (1987 September). Evaluation Of A Vector Quantization Talker Recognition System In Text Independent And Text Dependent Modes. *Pages 143–157 of: Computer Speech and Language*, vol. 22.

Rosenberg, A., Lee, C., & Soong, F. (1990 April). Sub-word Unit Talker Verification Using Hidden Markov Models. *Pages 269–272 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rosenberg, A., Lee, C-H., & Gokcen, S. (1991 May). Connected Word Talker Verification Using Whole Word Hidden Markov Model. *In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rosenberg, A., Delong, J., Lee, C-H., Juang, B. H, & Soong, F. K. (1992). The Use of Cohort Normalised Scores For Speaker Verification. *In: Proceedings of the International Conference on Spoken Language Processing.*

Rosenberg, A., Lee, C-H., & Soong, F. K. (1994). Cepstral Channel Normalization Techniques For HMM Based Speaker Verification. *Pages 1835–1837 of: Proceedings of the International Conference on Spoken Language Processing.*

Rosenberg, A. E. (1976 April). Automatic Speaker Verification: Review. *Pages 336–347 of: Proceedings of The IEEE*, vol. 64.

Rudasi, L., & Zahorian, A. (1991 May). Text Independent Talker Identification With Neural Networks. *Pages 389–392 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Rumelhart, D.E., & McClelland, J.L. (1986). *Parallel Distributed Processing, Exploration in the Microstructure of Cognition.* Vol. 1. Foundation MIT Press.

Sagisaka, Y. (1990 January). Speech Synthesis From Text. *Pages 35–41 of: IEEE Communications Magazine*, vol. 28.

Sakoe, H., & Iso, K. (1987). Dynamic Neural Network- A New Speech Recognition Model Based on Dynamic Programming and Neural Network. *In: IEICE Technical Report 87, NEC Corporation.*

Sakoe, H., Isotani, R., Yoshida, K., Iso, K., & Watanabe, T. (1989 May). Speaker-Independent Word Recognition Using Dynamic Programming Neural Networks. *Pages 29–32 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Sambur, M. (1976). Speaker Recognition Using Orthogonal Linear Prediction. *Pages 283–289 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP-24 no4.

Savic, M., & Gupta, S. (1990 April). Variable Parameter Speaker Verification System Based On Hidden Markov Modelling. *Pages 281–284 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Setlur, A., & Jacobs, T. (1995 September). Results Of A Speaker Verification Service Trial Using HMM Models. *Pages 639–642 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Sheikhzadegan, J., Tebiani, M., Lotfizad, M., & Roohani, M. (1995 September). Speaker Classification By Neural Networks For Short Utterances Using Phoneme Groups In Farsi. *Pages 375–376 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Shrimpton, D., & Watson, B. (1992 December). Comparison Of Neural Network Architectures For Speaker Verification. *Pages 460–464 of: Proceedings of the Australian International Conference on Speech Science and Technology*, vol. 1.

Soong, F., & Rosenberg, A. (1986). On The Use Of Instantaneous And Transitional Spectral Information In Speaker Recognition. *Pages 877–870 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Soong, F., Rosenberg, A., & Juang, L. Rabiner B. (1985 March). A Vector Quantization Approach To Speaker Recognition. *Pages 387–390 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Sutherland, A. (1989). *Automatic Speaker Verification Based On Waveform Perturbation Analysis*. Ph.D. thesis, University of Edinburgh.

Tabatabaee, V., Azimisadjadi, B., Zahirazami, S. H., & Lucas, C. (1994 April). Isolated Word Recognition Using Hybrid Neural Network. *Pages 649–652 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Takebayashi, Y., Tsuboi, H., & Kanazawa, H. (1991 May). A Robust Speech Recognition System Using Word-Spotting With Noise Immunity Learning. *Pages 905–908 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Thevenaz, P., & Hugli, H. (1994 April). Conformity, A New Method For Text-Independent Speaker Recognition. *Pages 63–66 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Tishby, N. (1991). On The Application Of Mixture AR Hidden Markov Models To Text Independent Speaker Recognition. *Pages 563–570 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. ASSP 30 no3.

Tobias, J. (1970). *Foundations of Modern Auditory Theory, Vol 1 and 2*. London and New York: Academic Press.

Trent, L., Rader, C., & Reynolds, D. (1994). Using Higher Order Statistics To Increase The Noise Robustness Of A Speaker Identification System. *Pages 221–224 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Tsoi, A. C., Shrimpton, D., Watson, B., & Back, A. (1994 April). Application Of Artificial Neural Network Techniques To Speaker Verification. *Pages 143–151 of: ESCA Workshop On Automatic Speaker Recognition Identification And Verification.*

Waibel, A., & Lee, K. (1990). *Reading in Speech Recognition.* Morgan Kaufmann Publishers, Inc.

Waibel, A., Sawai, H., & Shikano, H. (1989). Modularity And Scaling In Large Phonemic Networks. *Pages 1888–1898 of: IEEE Trans. on Acoustic, Speech, and Signal Processing*, vol. 37.

Wolf, J. (1972). Efficient Acoustic Parameters For Speaker Recognition. *Pages 2044–2056 of: The Journal of the Acoustic Society of America*, vol. 51 no:6.

Wu, F. H., Parhi, K., & Ganeesan, K. (1991 May). Neural Network Vector Quantizer Design Using Sequential And Parallel Learning Techniques. *Pages 637–640 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.*

Xu, L., & Mason, J. (1989 September). Instantaneous And Transitional Perceptually-Based Features In Speaker Identification. *Pages 271–274 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Xu, L., Oglesby, J., & Mason, J. (1989 May). The Optimization Of Perceptually- Based Features For Speaker Identification. *Pages 520–523 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 1.

Yang, J. (1993 April). Frequency Domain Noise Suppression Approaches In Mobile Telephone Systems. *Pages 363–366 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.

Yegnanarayana, B., Wagh, S., & Rajendran, S. (1994 September). A Speaker Verification System Using Prosodic Features. *Pages 1867–1870 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 4.

Yin, Hujun. (1990 November). Speaker Recognition Using Static and Dynamic Cepstral Feature By A Learning Neural Network. *Pages 1277–1280 of: Proceedings of the International Conference on Spoken Language Processing*, vol. 2.

Young, S., & Woodland, P. (1992). HTK: Hidden Markov Model Toolkit V1.4. *In: User Manual. Cambridge University Engineering Department, Speech Group*. Cambridge University.

Young, S. J. (1991 February). Competitive Training: A Connectionist Approach To The Discriminative Training of Hidden Markov Models. *Pages 61–68 of: IEE Proceedings*, vol. 138.

Yu, K., Mason, J., & Oglesby, J. (1995 September). Speaker Recognition Models. *Pages 629–632 of: Proceedings of the European Conference on Speech Technology*, vol. 1.

Zhu, M., & Fellbaum, K. (1990 May). A Connectionist Model For Speaker-Independent Isolated Word Recognition. *Pages 529–532 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 2.